Cryptographic Privacy-Preserving Enhancement Method for Investigative Data Acquisition

Zbigniew Kwecka

April 2011
Contents

CONTENTS .................................................................................................................................................. 2
LIST OF TABLES ....................................................................................................................................... 6
LIST OF FIGURES .................................................................................................................................... 7
ABSTRACT .................................................................................................................................................. 9
DECLARATION .......................................................................................................................................... 12
COPYRIGHT ............................................................................................................................................ 13
ACKNOWLEDGMENTS ............................................................................................................................ 14
LIST OF ACRONYMS ............................................................................................................................... 15

CHAPTER 1  INTRODUCTION .................................................................................................................. 17
  1.1 INTRODUCTION .................................................................................................................................. 17
  1.2 BACKGROUND AND CONTEXT ........................................................................................................ 18
  1.3 MOTIVATING SCENARIOS ............................................................................................................... 19
  1.4 RESEARCH QUESTION, AIM AND OBJECTIVES .............................................................................. 22
  1.5 CONTRIBUTION TO KNOWLEDGE ............................................................................................... 23
  1.6 PUBLICATIONS .................................................................................................................................. 24
  1.7 THESIS STRUCTURE ......................................................................................................................... 25

CHAPTER 2  BACKGROUND AND THEORY .............................................................................................. 27
  2.1 INTRODUCTION .................................................................................................................................. 27
  2.2 DIGITAL FORENSIC ............................................................................................................................ 27
  2.3 INVESTIGATIONS USING THIRD PARTY DATA ............................................................................... 31
    2.3.1 Data Protection Act ...................................................................................................................... 31
    2.3.2 Regulation of Investigatory Powers Act ...................................................................................... 33
    2.3.3 Data retention .............................................................................................................................. 35
    2.3.4 Commission and diligence for recovery of documents ............................................................. 37
2.4 PRIVACY, AND ITS WIDER CONTEXT ........................................................................ 37
   2.4.1 Privacy .............................................................................................................. 38
   2.4.2 Measuring privacy ............................................................................................ 39
   2.4.3 Privacy in surveillance systems .......................................................................... 40
2.5 CRYPTOGRAPHY .................................................................................................... 42
   2.5.1 Classification of cryptographic protocols .......................................................... 43
   2.5.2 Authentication, Integrity and Non-repudiation – Public Key Infrastructure (PKI) .... 45
   2.5.3 Operations on encrypted plaintext ...................................................................... 46
   2.5.4 Commutative Cryptography .............................................................................. 46
   2.5.5 Zero-Knowledge Proofs .................................................................................. 49
   2.5.6 Cryptanalysis .................................................................................................... 50
   2.5.7 Compliance ....................................................................................................... 52
   2.5.8 Trusted Computing (TC) .................................................................................. 53
2.6 CONCLUSION ......................................................................................................... 54

CHAPTER 3 LITERATURE REVIEW .............................................................................. 57
   3.1 INTRODUCTION .................................................................................................... 57
   3.2 PRIVACY-RESPECTING INVESTIGATIVE DATA ACQUISITION ................................ 58
      3.2.1 Privacy self-defence ...................................................................................... 58
      3.2.2 Privacy Controls in an Investigative Scenario .............................................. 60
   3.3 PRIVACY-PRESERVING PRIMITIVES .................................................................. 63
      3.3.1 Privacy-Enhancing Technologies ................................................................. 63
      3.3.2 Multi-Party Computation .............................................................................. 67
      3.3.3 Sharing a secret ........................................................................................... 67
      3.3.4 Retrieving records in a private manner ....................................................... 71
      3.3.5 Private Value Comparison – Locating interesting records ......................... 75
      3.3.6 Combined approaches to selective information retrieval ......................... 79
      3.3.7 Security Considerations .............................................................................. 82
   3.4 CONCLUSION ....................................................................................................... 85

CHAPTER 4 IMPROVING THE ACQUISITION PROCESS .............................................. 88
   4.1 INTRODUCTION .................................................................................................... 88
   4.2 METHODOLOGY .................................................................................................. 89
   4.3 INITIAL REQUIREMENTS .................................................................................... 90
   4.4 OVERALL DESIGN .............................................................................................. 91
   4.5 APPROACH 1: COMBINATION OF PET PRIMITIVES ........................................ 95
   4.6 APPROACH 2: COMBINED PET PRIMITIVES .................................................. 98
   4.7 EVALUATION ...................................................................................................... 99
      4.7.1 Experiment Design and Implementation ...................................................... 99
      4.7.2 Empirical Evaluation .................................................................................... 103
### 4.7.3 Feedback from practitioners

<table>
<thead>
<tr>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>109</td>
</tr>
</tbody>
</table>

### 4.8 REQUIREMENTS REVIEW

<table>
<thead>
<tr>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>110</td>
</tr>
</tbody>
</table>

### 4.9 CONCLUSIONS

<table>
<thead>
<tr>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>111</td>
</tr>
</tbody>
</table>

### CHAPTER 5 NOVEL DATA ACQUISITION FRAMEWORK

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 INTRODUCTION</td>
<td>113</td>
</tr>
<tr>
<td>5.2 METHODOLOGY</td>
<td>114</td>
</tr>
<tr>
<td>5.3 IDAP DESIGN</td>
<td>115</td>
</tr>
<tr>
<td>5.3.1 Lowering Processing Time</td>
<td>115</td>
</tr>
<tr>
<td>5.3.2 Allow multiple selection criteria</td>
<td>121</td>
</tr>
<tr>
<td>5.3.3 Reassuring the Public</td>
<td>122</td>
</tr>
<tr>
<td>5.4 IMPLEMENTATION</td>
<td>125</td>
</tr>
<tr>
<td>5.5 PROPOSED QUANTITATIVE EVALUATION</td>
<td>127</td>
</tr>
<tr>
<td>5.5.1 Overall design of experimental environment</td>
<td>127</td>
</tr>
<tr>
<td>5.5.2 Experiments</td>
<td>129</td>
</tr>
<tr>
<td>5.5.3 Proposed Qualitative Evaluation</td>
<td>131</td>
</tr>
<tr>
<td>5.6 CONCLUSION</td>
<td>132</td>
</tr>
</tbody>
</table>

### CHAPTER 6 EVALUATION

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 INTRODUCTION</td>
<td>134</td>
</tr>
<tr>
<td>6.2 PRESENTATION OF PERFORMANCE IMPACT</td>
<td>135</td>
</tr>
<tr>
<td>6.2.1 Evaluation of the complexity model</td>
<td>135</td>
</tr>
<tr>
<td>6.2.2 Varying the number of interesting records</td>
<td>139</td>
</tr>
<tr>
<td>6.2.3 Varying number of enquiries</td>
<td>142</td>
</tr>
<tr>
<td>6.2.4 Evaluation of IDAP with directory of identities</td>
<td>144</td>
</tr>
<tr>
<td>6.2.5 Use of dilution factor with different protocols</td>
<td>145</td>
</tr>
<tr>
<td>6.2.6 Controlling the balance between privacy and feasibility</td>
<td>146</td>
</tr>
<tr>
<td>6.3 PRESENTATION OF QUALITATIVE EVALUATION</td>
<td>147</td>
</tr>
<tr>
<td>6.3.1 Correctness and Security</td>
<td>147</td>
</tr>
<tr>
<td>6.3.2 Survey Results</td>
<td>150</td>
</tr>
<tr>
<td>6.4 CONCLUSION</td>
<td>151</td>
</tr>
</tbody>
</table>

### CHAPTER 7 CONCLUSIONS AND FUTURE WORK

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 INTRODUCTION</td>
<td>153</td>
</tr>
<tr>
<td>7.2 ACHIEVEMENT OF OBJECTIVES</td>
<td>154</td>
</tr>
<tr>
<td>7.3 MOTIVATING SCENARIOS WITH SOLUTIONS</td>
<td>158</td>
</tr>
<tr>
<td>7.4 CONTRIBUTION TO KNOWLEDGE</td>
<td>160</td>
</tr>
<tr>
<td>7.5 CRITICAL ANALYSIS</td>
<td>161</td>
</tr>
<tr>
<td>7.6 MAIN FINDINGS</td>
<td>165</td>
</tr>
<tr>
<td>7.7 FUTURE WORK</td>
<td>168</td>
</tr>
</tbody>
</table>
List of Tables

Table 4-1 Cryptographic operation performance measurements in nanoseconds (ns) ....................... 104
Table 4-2 Computational complexity of the OT-based approach.......................................................... 106
Table 4-3 Computational complexity of the PE-based approach.......................................................... 106
Table 6-1 Initial definition of IDAP’s complexity. .................................................................................. 137
Table 6-2 IDAP’s complexity. .............................................................................................................. 138
Table 6-3 IDAP’s complexity for datasets with publically available dictionaries. ............................... 139
List of Figures

<table>
<thead>
<tr>
<th>Figure Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Analogy to the operation of the three-pass protocol</td>
<td>48</td>
</tr>
<tr>
<td>2-2</td>
<td>Three-pass protocol operation</td>
<td>48</td>
</tr>
<tr>
<td>2-3</td>
<td>Illustration of Zero-Knowledge-Proofs</td>
<td>49</td>
</tr>
<tr>
<td>2-4</td>
<td>Simplified Explanation of Zero-Knowledge-Proofs</td>
<td>50</td>
</tr>
<tr>
<td>3-1</td>
<td>PET Staircase: the effectiveness of the different conventional PET types</td>
<td>66</td>
</tr>
<tr>
<td>3-2</td>
<td>Locking a secret under Khayat’s secret sharing scheme</td>
<td>68</td>
</tr>
<tr>
<td>3-3</td>
<td>Illustration of locking a secret under Khayat’s secret sharing scheme</td>
<td>69</td>
</tr>
<tr>
<td>3-4</td>
<td>Illustration of unlocking a secret under Khayat’s secret sharing scheme</td>
<td>69</td>
</tr>
<tr>
<td>3-5</td>
<td>Unlocking a secret under Khayat’s secret sharing scheme</td>
<td>70</td>
</tr>
<tr>
<td>3-6</td>
<td>Shamir’s simple secret sharing scheme</td>
<td>71</td>
</tr>
<tr>
<td>3-7</td>
<td>Basic Oblivious Transfer Protocol by Schneier</td>
<td>73</td>
</tr>
<tr>
<td>3-8</td>
<td>Description of Private Equality Test based on Commutative Cryptography</td>
<td>77</td>
</tr>
<tr>
<td>3-9</td>
<td>Private Equality Test based on Commutative Cryptography</td>
<td>77</td>
</tr>
<tr>
<td>3-10</td>
<td>Description of 1-n Private Equality Test protocol based on Commutative Cryptography</td>
<td>79</td>
</tr>
<tr>
<td>3-11</td>
<td>1-to-n Private Equality Test protocol based on Commutative Cryptography</td>
<td>79</td>
</tr>
<tr>
<td>3-12</td>
<td>Operation of the Private Equi-join protocol</td>
<td>81</td>
</tr>
<tr>
<td>3-13</td>
<td>Efficient PIR based on Secure Coprocessor</td>
<td>81</td>
</tr>
<tr>
<td>4-1</td>
<td>Typical request for investigative data mapped into SQL</td>
<td>92</td>
</tr>
<tr>
<td>4-2</td>
<td>Request enabling privacy-preserving queries mapped into SQL</td>
<td>93</td>
</tr>
<tr>
<td>4-3</td>
<td>Retrieval Phase</td>
<td>98</td>
</tr>
<tr>
<td>4-4</td>
<td>Test Dataset</td>
<td>101</td>
</tr>
<tr>
<td>4-5</td>
<td>Total running time for both approaches including preparation time</td>
<td>107</td>
</tr>
<tr>
<td>4-6</td>
<td>Data Acquisition processing time excluding preparation time</td>
<td>107</td>
</tr>
<tr>
<td>4-7</td>
<td>Performance of both approaches for varied number of records (1 &lt; m &lt; n)</td>
<td>108</td>
</tr>
<tr>
<td>5-1</td>
<td>Lowering Processing Time Phase A – Preparation</td>
<td>118</td>
</tr>
<tr>
<td>5-2</td>
<td>Lowering Processing Time Phase B – Searching</td>
<td>118</td>
</tr>
<tr>
<td>5-3</td>
<td>Lowering Processing Time Phase C – Retrieval</td>
<td>120</td>
</tr>
</tbody>
</table>
Figure 5-4 Process flow of the protocol incorporating the *dilation factor* ........................................... 120
Figure 5-5 mapped into SQL.................................................................................................................. 121
Figure 5-6 Reassuring the public by introducing semi-trusted third parties........................................ 124
Figure 5-7 IDAP...................................................................................................................................... 126
Figure 5-8 Measuring processing time .................................................................................................. 128
Figure 5-9 Measuring bandwidth used during a protocol run ............................................................... 129
Figure 6-1 Complexity table reading vs. actual measurements............................................................... 136
Figure 6-2 IDAP complexity in detail for varied *m; k=1; o=1,000 and n=1,000,000* ......................... 138
Figure 6-3 Computational complexity of IDAP and PE for increasing *m* (logarithmic scale)............. 140
Figure 6-4 Computational complexity of IDAP and PE for increasing *m* ............................................. 141
Figure 6-5 IDAP’s processing time for varying *m* and different values of *n* ....................................... 141
Figure 6-6 IDAP’s processing time for varying *m* and different values of *o* ..................................... 141
Figure 6-7 Comparison of processing time for IDAP and PE............................................................... 142
Figure 6-8 Detailed comparison for IDAP and PE .................................................................................. 143
Figure 6-9 IDAP’s performance for different values of *γ*, as compared to PE..................................... 143
Figure 6-10 Performance gain for IDAP run on dataset with a directory............................................. 144
Figure 6-11 Performance gain for IDAP run on dataset with a directory............................................ 145
Figure 6-12 Comparison of IDAP to a modification of OT-based approach ........................................ 146
Figure 6-13 Processing time against the dilution factor *o*, for IDAP with directory of records........... 147
Figure 7-1 Hiding of request originators can improve privacy .............................................................. 166
The current processes involved in the acquisition of investigative data from third parties, such as banks, Internet Service Providers (ISPs) and employers, by the public authorities can breach the rights of the individuals under investigation. This is mainly caused by the necessity to identify the records of interest, and thus the potential suspects, to the dataholders. Conversely, the public authorities often put pressure on legislators to enable them a more direct access to the third party data, mainly in order to improve on turnaround times for enquiries and to limit the likelihood of compromising the investigations. This thesis presents a novel methodology for improving privacy and the performance of the investigative data acquisition process.

The thesis shows that it is possible to adapt Symmetric Private Information Retrieval (SPIR) protocols for use in the acquisition process, and that it is possible to dynamically adjust the balance between the privacy and performance based on the notion of $k$-anonymity. In order to evaluate the findings an Investigative Data Acquisition Platform (IDAP) is formalised, as a cryptographic privacy-preserving enhancement to the current data acquisition process.

SPIR protocols are often computationally intensive, and therefore, they are generally unsuitable to retrieve records from large datasets, such as the ISP databases containing records of the network traffic data. This thesis shows that, despite the fact that many potential sources of investigative data exist, in most cases the data acquisition process can be treated as a single-database SPIR. Thanks to this observation, the notion of $k$-anonymity, developed for privacy-preserving statistical data-mining protocols, can be applied to the investigative scenarios, and used to narrow down the number of records that need to be processed by a SPIR protocol.
This novel approach makes the application of SPIR protocols in the retrieval of investigative data feasible.

The *dilution factor* is defined, by this thesis, as a parameter that expresses the range of records used to hide a single identity of a suspect. Interestingly, the value of this parameter does not need to be large in order to protect privacy, if the enquiries to a given dataholder are frequent. Therefore, IDAP is capable of retrieving an interesting record from a dataholder in a matter of seconds, while an ordinary SPIR protocol could take days to complete retrieval of a record from a large dataset.

This thesis introduces into the investigative scenario a semi-trusted third party, which is a watchdog organisation that could proxy the requests for investigative data from all public authorities. This party verifies the requests for data and hides the requesting party from the dataholder. This limits the dataholders ability to judge the nature of the enquiry. Moreover, the semi-trusted party would filter the SPIR responses from the dataholders, by securely discarding the records unrelated to enquiries. This would prevent the requesting party from using a large computational power to decrypt the diluting records in the future, and would allow the watchdog organisation to verify retrieved data in court, if such a need arises. Therefore, this thesis demonstrates a new use for the semi-trusted third parties in SPIR protocols. Traditionally used to improve on the complexity of SPIR protocols, such party can potentially improve the perception of the cryptographic trapdoor-based privacy-preserving information retrieval systems, by introducing policy-based controls.

The final contribution to knowledge of this thesis is definition of the process of privacy-preserving matching records from different datasets based on multiple selection criteria. This allows for the retrieval of records based on parameters other than the identifier of the interesting record. Thus, it is capable of adding a degree of fuzzy matching to the SPIR protocols that traditionally require a perfect match of the request to the records being retrieved. This allows for searching datasets based on circumstantial knowledge and suspect profiles, thus, extends the notion of SPIR to more complex scenarios.

The constructed IDAP is thus a platform for investigative data acquisition employing the Private Equi-join (PE) protocol – a commutative cryptography SPIR protocol.
The thesis shows that the use of commutative cryptography in enquiries where multiple records need to be matched and then retrieved ($m$-out-of-$n$ enquiries) is beneficial to the communicational and computational performance. However, the above customisations can be applied to other SPIR protocols in order to make them suitable for the investigative data acquisition process. These customisations, together with the findings of the literature review and the analysis of the field presented in this thesis, contribute to knowledge and can improve privacy in the investigative enquiries.
Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institution of learning.
Copyright

Copyright in text of this thesis rests with the Author. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the Author and lodged in the Napier University. Details may be obtained from the Librarian. This page must form part of any such copies made. Further copies (by any process) of copies made in accordance with such instructions may not be made without the permission (in writing) of the Author. The ownership of any intellectual property rights which may be described in this thesis is vested in Napier University, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the University, which will prescribe the terms and conditions of any such agreement. Further information on the conditions under which disclosures and exploitation may take place is available from the Head of School of Computing.
Acknowledgments

First, and foremost, I would like to thank my family for never doubting in me. So to my wife Wiola, to my parents Maria and Roman, and to my brother Andrzej, I thank you for always being there for me. Also, thanks to my daughter Amelia for keeping me sane near the end of this lengthy process, as her smile can always take weight off my shoulders.

My supervision team has always provided support and suggested plenty of avenues to explore. Prof. William Buchanan, my supervisor, was always an inspiration and never-ending source of research ideas. To Duncan Spiers, the second supervisor, many thanks for discussing the legal aspects of this thesis with me and for helping me to validate my ideas. Dr Michael Smyth has been a great panel chair, as keen listener with an eye for detail he has kept me on the right track throughout the process.

Finally, huge ‘thank you’ to my friends and colleagues. With special thanks to the C44 research team: to Jamie, Kevin, Lionel and Tim for getting me interested in doing research and helping me with my first steps. Also, I would like to thank Stuart and Allan for reading things at the last minute and providing ideas for improvements.

I know I did not manage to mention you all in here, but I do remember your efforts in motivating me. So thanks to all that throughout this long process have asked me a countless number of times “So when are you getting your PhD?”. You know what? I just wouldn’t make it without you!

Thank You!
List of Acronyms

CSP  Content Service Provider
FIPS  Federal Information Processing Standard
IDAP  Investigative Data Acquisition Platform
MPC  Multi-Party Computation
OT  Oblivious Transfer
PE  Private Equi-Join
PET  Privacy Enhancing Technology
PIR  Private Information Retrieval
SCOP  Secure Coprocessor
SIPR  Scottish Institute for Policing Research
SPIR  Symmetric PIR
SPoC  Single Point of Contact
TC  Trusted Computing
TOR  The Onion Router
ZKP  Zero-Knowledge Proof
Those who would give up essential Liberty, to purchase a little temporary safety, deserve neither Liberty nor Safety.

-Benjamin Franklin, 11 Nov 1755
Chapter 1

Introduction

1.1 Introduction

This thesis provides a novel method for obtaining investigative data from third-parties. Current processes often breach the human rights of the individuals being investigated, and they may expose the investigation. Consequently, in this thesis, a Privacy Enhancing Technologies (PET) solution is proposed based on the Symmetric Private Information Retrieval (SPIR) primitive customised to perform the specific task of data acquisition. The proposed customisation is the main contribution to knowledge of this thesis, and it can be applied to most single database SPIR protocols.

This research commenced a few years after the tragic events of 9/11, which started the 21st Century’s War on Terror. It was this event that caused the UK to alter the laws and procedures for obtaining intelligence data towards more intrusive solutions. Thus, despite the protests of civil liberty activists, the UK decision-makers are still
prepared to widen the warrantless access to data, in order to satisfy the needs of investigators in the ever-changing communication environment. The possibility of direct access of the public authorities to information held by Communication Service Providers (CSPs) is one of the fiercely discussed topics. Consequently, the aim of this thesis is to propose a privacy-respecting mechanism that could be used by investigators in order to access data held by third-parties.

This thesis, thus, identifies that a major issue in the way that investigations are conducted is the need for public authorities to disclose the identities of their suspects to the data-holders. Such a disclosure may negatively impact the suspects, and also jeopardise the investigations. In order to protect the interest of investigations the public authorities may be forced to wait before they have a number of similar cases, or to widen their enquiry, in order to dilute suspect's identity. This, depending on which technique is used, can lead to delays, or be treated as fishing-for-evidence. Therefore, this thesis sets out to provide an efficient way to automate the investigative data acquisition process, if the rights of the data subjects and the secrecy of the investigations are to be protected. The solution should be able to perform tasks regulated by a number of, often contradicting, legislative requirements. It should also allow for the fine tuning of the correct balance between privacy, security and performance.

1.2 Background and Context

Since 11 September 2001 many Western governments have passed laws empowering public authorities with wider rights to gather operational data [1, 2]. For many years public opinion accepted the invasion of personal privacy rights as the sacrifice needed to fight terrorism [3]. However, slowly, public opinion is shifting back to a state where such measures are often considered unacceptable. This is shown by public opinion surveys, such as the one conducted in US by Washington Post [4], where 32% of respondents agreed that they would prefer the federal government to ensure that privacy rights are respected rather than to investigate possible terrorist threats. This was an 11% increase from the similar survey conducted in 2003. The trend continues, since in [5] 63% of respondents stated that they are worried about the government’s surveillance into personal lives.
In the UK, the public authorities, including the Police, request investigative data from third-parties on regular basis [6] and the data protection legislation allows for such requests, even without warrants [7, 8]. Depending on the way these requests are performed, human and natural rights of the data-subject can be breached, and/or the investigation may be compromised [9]. A recent proposal by the UK government went further and recommended allowing the public authorities direct access to data held by Content Service Providers (CSPs), such as mobile telephony providers, and Internet Service Providers (ISPs) [2]. There are a few major motivating factors behind this proposal: increasing access speeds to records; allowing for covert enquiries by anti-terror and national security agencies; reducing collateral damage to potential suspects under investigation; and enabling the analysis of data to facilitate the profiling of terrorists activities. In response, concerns were raised that if the proposal was implemented, it would thwart the privacy of Internet users around the globe, in order to increase the security of one nation. This thesis shows that most of the objectives set out in the proposal can still be achieved while maintaining a high level of privacy. It is shown that an investigative system can maintain the privacy of the data subjects and also preserve the confidentiality of investigations. However, both security and privacy must be built into the system at the design stage in order to achieve this [1].

This thesis, thus, gives an insight into the use of PETs in improving the current investigative data acquisition practices and defines the Investigative Data Acquisition Platform (IDAP). IDAP is a proposed novel approach to maintain secrecy; preserving the suspect’s privacy and gaining the public’s support for the PET technologies in digitalised investigative enquiries.

1.3 Motivating Scenarios

There are a number of possible scenarios that the data acquisition process needs to facilitate. Legislations permit different public authorities to request investigative data from third parties under a variety of circumstances. To highlight the possible breaches of privacy and/or human rights of the suspects, we consider the following scenarios:
Scenario 1: Request for ISP subscriber data
A forensic investigation, carried out by a law enforcement agency on a confiscated Personal Computer (PC), has identified 14 different Internet Protocol (IP) addresses linked to organised crime. The agency would like to identify the owners of these IP addresses, and their subscriber data including the postal address. However, it is key that the nature of the enquiry, and identities of sought after individuals, are not revealed to the ISP (directly, or indirectly) in order to protect the integrity of the investigation.

Scenario 2: Banking transaction details
A shopkeeper has notified a law enforcement agency about the purchase of an uncontrolled substance that, in the wrong hands, can be used to produce an exploding device. The credit card number used in the transaction is made available to the agency. This agency would like to find out the list of purchases made on this card for the previous month, as well as the name of the owner. Since banks are not obliged to provide such information to the public authorities, the nature of the enquiry will have to be communicated to the bank. However, the identity of the potential suspect should be kept secret from the bank, as not to affect this individual’s relation with the bank. If the bank was aware of a given individual being a suspect in an investigation then, as an example, the individual could be placed on a list of high-risk borrowers. This may stop them from getting a loan, even though they have not been charged with a crime. Most importantly, this individual may be unable to find out why his application was refused, since the disclosure of matters affecting national security and crime prevention are exempt from many provisions of the UK’s Data Protection Act 1998 (DPA) (sections 28 and 29)[10].

It is clear that in these scenarios, investigations can be compromised by revealing the identity of the suspects to the data-holders. However, the second scenario demonstrates more clearly an invasion of suspect’s rights, and, in this case, the party that caused the violation is the security services, as their actions have made a third-party aware of the identity of a suspect in an investigation.

In the UK, according to the DPA, organisations may provide other organisations with personal and sensitive personal information about a data subject in some exceptional circumstances (see Part IV of the DPA [10]). For instance, emergency services may
request information on the allergies of a casualty, and of a casualty’s relatives, from any organisation that they suspect may have this data, and such organisation may lawfully disclose the data. Accordingly, the police and other public authorities may also request data related to their suspects, based on the same reasoning. Thus, in Scenario 2, the security services and the data-controlling organisation would act lawfully in accordance with the above legislation. However, their actions could seriously impinge upon the data-subject’s natural rights, and, quite possibly, their right to privacy. In this scenario, it could have a detrimental impact upon the data-subjects rights concerning the future relations with the data-controlling organisation.

This raises interesting issues about the legal remedies, if any, open to the suspect. In similar circumstances a case could be made that there has been a breach of Article 8 of the Council of Europe’s Convention on Human Rights (now enforceable in the UK under the Human Rights Act 1998 [11]). This would be difficult to pursue for a number of reasons. Quite apart from the practical difficulty of knowing that there has been a breach of rights, how the breach has come about, who is responsible and how to prove it (what might be called evidential difficulties), there is also the question of the extent to which those responsible might be able to claim exemption from responsibility (which might be called substantive difficulties). The right of privacy under Article 8 (like most human rights) is a qualified right, meaning that a public authority is entitled to disregard the right where the interests, among others, of national security, or the prevention of crime and disorder, require. Such an exemption would normally exclude the possibility of the affected data subject being able to pursue damages against the public authority. However, perhaps the correct approach is to regard the exemptions as only coming into effect where they are proportionate. If there is a way to obtain the evidence they require without invasion of privacy and other rights of the suspect, and without the adverse impact the scenario predicts, it is arguable that the public authority should take into account the rights of the suspect, and so to choose the least disruptive method of obtaining the evidence they need. It therefore could be argued that if they chose a method, which invades protected rights, and is likely to cause adverse impacts, the public authority have used an exemption disproportionately, and so should be obliged to recompense the suspect for the harm perpetrated by their choice of method. It is interesting to conjecture to what extent a court would entertain such a claim.
1.4 Research Question, Aim and Objectives

The work presented in this thesis will therefore address the following research question:

What are the improvements to SPIR methods that can be made within an investigative framework, and how can this be evaluated against the current methods to show efficiency gains?

Consequently, the main aim of this research is as follows:

To define new methods for the investigative data acquisition that can preserve privacy of relevant data-subjects, and which have perceivable performance gains on existing methods, and to allow variable parameters to preserve the balance of privacy against performance.

The thesis addresses these considerations with the following objectives:

1) Construct a literature review within the PET sphere (Chapter 3).

2) Define a set of requirements that data acquisition process must meet (Chapter 4).

3) Construct a novel methodology for privacy-preserving investigative data acquisition (Section 5.3 and Section 5.4).

4) Propose an evaluation framework suitable to assess performance of novel cryptographic enhancements to retrieval of investigatory data (Section 5.5).

5) Investigate parameters that could be used to assess the balance between the privacy and feasibility (Section 6.2.6).

In theory, it is possible to use Private Information Retrieval (PIR) primitives to search databases belonging to third parties, without revealing the search criteria. Thus, in investigative data acquisition, it is feasible to keep the identity of the suspects secret. While a PIR protocol would reveal to the investigators other records than those classified as interesting (the records referring to the suspect), a SPIR
protocol would potentially protect the interests of all the parties involved, since it can:

- Protect an enquiry by hiding the identities of the interesting records and some of the search criteria.
- Protect records kept on database, but unrelated to the enquiry.

However, even efficient SPIR protocols may struggle to handle privacy-preserving requests for investigative data, as the databases involved usually contain a large number of records that are likely to change frequently. Consequently, it will be necessary to investigate possible modifications to existing SPIR protocols that would be suitable for enhancing the performance of these protocols in an investigative scenario. At the same time, the balance between the performance and the privacy must be kept at acceptable level. Therefore, the criteria for selecting this acceptable level will be also defined by this thesis.

The main goals of this work are to put forward a new problem, establishing a practical feasibility result for this problem, and, in the process, develop techniques that allow for the scaling-up current SPIR schemes to the size required by investigative data acquisition. This work does not attempt to fully optimise the platform, as this would complicate the presentation of the problem and the solution. The platform should, thus, be mainly viewed as a feasible framework, which may be the basis for further optimisations.

### 1.5 Contribution to Knowledge

The work presented in this thesis contributes the following to knowledge:

1) Demonstrating the manner in which SPIR techniques can be used to assist public authorities in privacy-preserving retrieval of investigative data from third parties.

2) Reducing the problem of investigative data acquisition to a single-database SPIR, thus, allowing for the limiting of the number of records that need to be collected from a dataholder in order no to affect the privacy of a suspect in a considerable way.
3) Presentation of a novel methodology for the privacy-preserving investigative data acquisition, IDAP, which is suitable for real-life implementation.

4) Creation of a dilution factor that can be used to control the balance between the privacy and performance in a single-database SPIR system.

5) Definition of a technique for building complex privacy-preserving enquiries based on multiple selection criteria.

6) The novel use of semi-trusted third parties to gain the support of the public for SPIR-based data acquisition techniques.

1.6 Publications

Appendix E presents main publications conducted with this research, which include:


- Conference paper presented at IEEE Region 8 International Conference on Computational Technologies in Electrical and Electronics Engineering - SIBIRCON (Irkutsk, 2010), entitled Privacy-Preserving Data Acquisition Protocol [13].


- Conference paper presented at 8th ECIW (Lisbon, 2009), entitled Application and Analysis of Private Matching Schemes Based on Commutative Cryptosystems [14].

- Research poster Privacy-Preserving Investigations - A technical solution to allow for legal and ethical data sharing that was presented during the second
1.7 Thesis Structure

The structure of this manuscript closely follows the methodology used to draw the final conclusions. Therefore, along with appendices that contain supporting data, this thesis is organised as:

- **Chapter 1 – Introduction.** This chapter outlines a brief context of the research domain, and identifies the key issues. Finally, it presents the research question to be explored.

- **Chapter 2 – Background and Theory.** This chapter provides a more detailed presentation of the background of investigative data acquisition and of privacy issues in information systems. It also provides an insight into cryptography, which forms the basis of the privacy-preserving data acquisition technique presented in this thesis.

- **Chapter 3 – Literature Review.** This chapter presents an analysis and overview of privacy-preserving techniques that can, in theory, improve the ethics of investigative data acquisition.

- **Chapter 4 – Improving the Acquisition Process.** This chapter defines the requirements for investigative data acquisition process and analyses existing PETs in contrast to these requirements. As a result, a single protocol is selected as a candidate protocol for the acquisition process and presented, along with a number of drawbacks of this protocol.

- **Chapter 5 – Novel Data Acquisition Framework.** This chapter presents the framework developed during this research, in order to explore the research question presented in Chapter 1. This framework is based on the PET protocol chosen and customised within this thesis for the specific task of investigative data acquisition. The chapter also outlines the process of experimentation and simulation, and provides the necessary narrative to place it within a research methodology.
• **Chapter 6 – Evaluation.** This chapter presents the results of the process of experimentation and simulation of the framework performance. It also includes the details on the qualitative evaluation of the framework.

• **Chapter 7 – Conclusions and future work.** This chapter provides a discussion and a summary of the main findings of this thesis, within the main considerations of the research domain. It will also justify the contributions to knowledge, and suggest the future work.
Chapter 2

Background and Theory

2.1 Introduction

This chapter introduces the concepts that are crucial to understanding the field of investigative data acquisition, and associated issues. Consequently, matters relating to the use of electronic evidence and obtaining investigative data from third parties are discussed below. This is followed by an introduction to the concepts of security and privacy in the information systems. Finally, different cryptographic techniques that find use in information retrieval and storage are outlined.

2.2 Digital Forensic

This section provides the background on the field of digital forensics. The definition of the term is followed by the comparison of the digital forensic to forensic science. This is done in order to introduce the concepts of investigation, and digital evidence, which includes data collected during investigations.
The most comprehensive definition of digital forensic reads as follows:

*(DF is) the use of scientifically derived and proved methods towards the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations.*

*Palmer, [15], pp. 16*

Computer-related offences started to emerge when personal computers became easily accessible. Nevertheless, DF was only recently recognised as a separate discipline. While technology is now advanced enough to assist in DF investigations, none of the DF techniques used is court-approved in the UK. This means that the DF evidence cannot be presented directly in court, unlike the physical evidence, such as the results of the fingerprints analysis. Instead, most DF evidence needs to be interpreted and presented to court by an expert witness in order to have any significance in a court case. Also, the lack of proper regulation of DF means that on many occasions electronic evidence is rejected by courts due to its illegitimate origin or alleged mishandling [16].

Little can be done to improve the quality of DF evidence collected from personal computing devices, as the environment is under a full control of the end-user. However, legislations for the corporate world, corporate information systems, as well as communications links, could allow for a greater access to data for investigators [16]. Currently, in order to protect themselves from costly legal suits, organisations often choose to monitor and log the flow of information that occurs under their governance. However, the methods that are used in these activities can render any information that is provided to the investigators, unusable in court. This is partially due to the fact that often the data is logged without the clear and explicit consent of the sender, or recipient, of the data, where by British and European law interception of information transfer without such consent is forbidden [17]. There are also more difficulties in using data from the corporate environment as evidence, as in cases
involving prosecution evidence provided by commercial organisations and their employees, the organisations can be selective about the evidence they provide [16].

During conventional forensics investigations, scientists are often capable of reconstructing events by examining physical evidence; therefore, it should also be possible in the computer environment. However, electronic evidence is quite different from the physical evidence. Traces of evidence in conventional forensic science are often difficult to forge; however, this is not the case in the world of digital forensics. For example, perpetrators can put fake evidence onto a machine belonging to someone else, or claim they become have victims of a Trojan horse[18]. This is partially due to the fact that conventional techniques of storing information are often more tamper-proof than their digital equivalents. For instance, if someone were to erase and overwrite a message written on paper, this process is likely to leave evidence of the information been tempered with. A perpetrator, thus, with moderate skills would be capable of a similar forgery in a digital world without leaving traces of this activity [19]. Within a computer operating system there are certain controls that can be used to trace such actions (such as meta-data, event logs, and so on), but these can also be fooled [18]. Some events, though, that take place in information systems often do not leave any long-lasting physical traces, as they only run in volatile memory [20, 21]. Thus, while monitoring computer systems, as well as in communication links, it can be treated as a potential breach of privacy and it can be used to prosecute the guilty, but it also have a potential to protect the innocent.

During a DF investigation, an examination of the data can also show signs of incidents different to the one being investigated. Thus, handling and examination of the evidence should be performed in a way defined by the appropriate regulatory documents [20]. In the corporate environment, these documents would most likely be a part of institutional security policy, written in accordance to the guidelines provided by national law enforcement. The UK authority, Association of Chief Police Officers (ACPO), provides general guidance for the recovery and handling of digital evidence. They suggest four following principles in working with computer-based electronic evidence (ACPO [20], pp. 4):

- No action taken ... should change data held on a computer or storage media which may subsequently be relied upon in court.
• In exceptional circumstances, where a person finds it necessary to access original data ..., that person must be competent to do so and be able to give evidence explaining the relevance and the implications of their actions.

• An audit trail or other record of all processes applied to computer-based electronic evidence should be created and preserved. An independent third party should be able to examine those processes and achieve the same result.

• The person in charge of the investigation has overall responsibility for ensuring that the law and these principles are adhered to.

The common practice of preserving digital evidence is taking complete copies of the source data bit-by-bit using raw formatting [22]. This, though, is often a costly and wasteful approach, as, in order to create forensically sound copy of a 200GB hard disk requires 200GB of storage for the evidence, even if there is only 100MB of data on the drive. The introduction of a vendor independent Common Digital Evidence Storage Format is discussed in [22], which allows data from various sources, including hard drives, network traffic monitoring, memory dumps and other files, or data acquired as evidence, to be preserved in a single format. This is the digital equivalent of the evidence bags that are used for physical evidence. Additionally, [23] suggests that evidence derived from server logs and network probes, such as the traffic data collected by the ISPs, should be split into different data formats, and data that is repeated should not be duplicated in the storage. In practice, this would mean that timestamps are stored in a date format, while IP addresses could be stored within 32 bits (four bytes), rather than in the decimal-dot-delimited format that requires up to 15 bytes when stored in raw format as a string. Also, the IP addresses could be stored for the initial packets in a given TCP connection logged, but then omitted from the remaining log entries. However, these considerations are mostly important to the dataholders. Since this thesis focuses on the mechanism for retrieving small quantity of records from datasets held by third parties, this in not an issue, but it is worth noting that there are valid concerns related to the storage requirements for safekeeping any potential evidence.
2.3 Investigations using third party data

The public authorities are often required to carry out investigations based on data supplied by third parties. Such investigations may include: benefit fraud enquiries from the HMRC; solving a crime by the Police; investigating alleged terrorism cases by Scotland Yard; or, gathering health information about a patient at an Accident and Emergency (A&E) department. The process of obtaining third-party records is usually referred to as data acquisition [8].

In the UK there are two major legislations that the public authorities can use to justify their request for third-party data. These are The Data Protection Act 1998 (DPA) [10], and The Regulation of Investigatory Powers Act 2000 (RIPA) [17], including its Scottish counterpart The Regulation of Investigatory Powers (Scotland) Act 2002 (RIPSA). In this thesis RIPA and RIPSA are both referred to as RIPA, unless specified differently. The reason for the lack of the distinction is that the matters relating the data acquisition are mostly the same for the UK. This section discusses the aspects of DPA and RIPA that are important to the data acquisition process, along with current data retention practices by Content Service Providers (CSPs).

2.3.1 Data Protection Act

In the UK, the DPA regulates the processing of data on identifiable individuals. Similar regulations also exist in other countries of the European Economics Area, as the DPA was enacted in implement of the European Data Protection Directive 95/46/EC [7]. The DPA provides eight principles for handling personal data and ten conditions governing the processing of sensitive personal data. It should be noted that this legislation is not aimed at regulating all data about individuals, and the scope of principles provided is limited by the definitions of data and sensitive personal data provided in Section 1(1) of the Act. Consequently, the DPA regulates the processing of any data about an individual, which is:

- intended to be processed automatically;
• intended to be recorded in a structured manner allowing for the retrieval of information about an identifiable individual, i.e. as a part of relevant filling system;

• a relevant health, educational or public record.

Additionally, the key to understanding DPA are the terms data subject and processing. Data subject is a term widely used to describe an identifiable individual whose data is kept by the given dataholder (referred to as data controller in the Act), where processing is used to describe any operation on the data.

The eight principles of the DPA are:

• Personal data shall be processed fairly and lawfully. Thus, any operation on the data relating to a data subject is done with expressed or implied consent of the data subject, unless it is required to satisfy legal requirements of the data controller.

• Information should only be used for the original purpose for which it has been obtained.

• Personal data needs to be adequate, relevant, and not excessive to the purpose for which they are processed.

• Data controller needs to make sure that the data is accurate and kept up-to-date when necessary.

• Data should be kept on a system only when needed for the original purpose for which it has been obtained.

• Data subject should be assured that any processing of personal data is performed in accordance to the DPA.

• Data controllers should regularly evaluate the risk to data and implement appropriate countermeasures if required (please see Appendix A for more information about assessing and mitigating risk).
The data cannot be transferred to a territory outside the European Economic Area (EEA) unless this territory can provide adequate level of protection for the data. Since, DPA and similar legislations guarantee to protect the rights of the subject only in the EEA, care needs to be taken when data are transferred to territories outside of these controls.

The DPA provides a voluntary mechanism to enable the data controller to disclose information on a data subject to the public authorities, in circumstances that the data controller perceives as reasonable for such disclosure. Some may argue that such an exclusion in the DPA is unreasonable, since the public authorities can then obtain information about their suspects without any court warrants. On the other hand, such a provision is required for life-threatening situations where medical staff or police needs to gather information quickly to protect lives and critical infrastructure. Consequently, the valid uses of this provision possibly outweigh the abuses. Based on the DPA, the public authorities cannot enforce any disclosure of information without a warrant. However, the data controller can disclose any information to investigators, if investigators can demonstrate a valid reason for the disclosure.

An example of when the system was abused is found in [24] where an ex-policeman was able to gain access to the database of UK Driver and Vehicle Licensing Agency (DVLA) and obtain postal address of an individual based on registration of a car. The ex-serviceman performed this action in order to help in his private investigation for a missing dog. This was a plain breach of the DPA, however, the consequences of this breach were limited, as if the missing dog enquiry was handled through the police channels, it would result in the same information being obtained by the ex-policeman. On the other hand, the DPA voluntary disclosure can be used to obtain a data subject's medical details by a hospital A&E department. This can allow the public authorities to act fast in life and death situations, where even a slight delay may cost the data subject, or another individual, their life.

2.3.2 Regulation of Investigatory Powers Act

CSPs are the most common third party sources of investigative data [25]. Historically CSPs stored communication data of all transmissions taking place for the billing purposes. Such data, often referred to as traffic data, included telephone numbers,
time of call, duration, and so on. ISPs, a subset of CSPs, used to record similar data for the Internet transaction, i.e. IP addresses; types of packets; some high-level Internet addresses, such as Hyper-Text Transfer Protocol (HTTP) Uniform Resource Locators (URLs). Consequently, in the past, investigators were able to make enquires requesting details of communication data based on the voluntary provision mechanism of the DPA. However, ambiguity and abuse of this investigative technique showed the need for further regulation of this area [26]. Consequently, RIPA was introduced to regulate:

- amount of information collected by CSPs about their customers;
- amount of time CSPs were allowed to retain this data;
- who, and under what circumstances, can request to see this data without a subpoena.

The rules addressing these issues are only a part of the RIPA act that was introduced in order to satisfy the directives of the European Convention on Human Rights [27]. RIPA set out to control the interception of communications, acquisition and disclosure of: communications data; the use of cover surveillance and human intelligence sources; as well as access to electronic data protected by passwords. Soon after RIPA, its Scottish counterpart RIPSA was announced, and regulates the general conduct of surveillance in Scotland [28]. According to the Home Office, these acts were supposed to strictly limit the use of covert surveillance techniques, and intrusive intelligence gathering to the most serious crimes, and thus, provide an independent body for investigating complaints from the public. However, it was criticised by lawyers and privacy activists for loosely defining what was meant by the most serious crimes, and the exceptional circumstances that allowed the public authorities a warrantless intrusion of privacy [25]. Although, the history shows that, generally, the rights given to the public authorities by RIPA are not being abused [28].

Under RIPA, a public authority may send a data acquisition notice to a CSP requesting the disclosure of certain traffic data. Unlike, in the DPA data acquisition request, RIPA notices do not require justification being presented to the data controller. The data controller must then disclose the requested data within a
reasonably practicable time, or face a penalty. Since, RIPA requires the collection of a relevant subset of data from a database of the CSP, the requesting party should make a financial contribution to cover the costs incurred by the CSP. All the notices are then subject to the approval by the senior officers of the requesting party. In the case of police, a RIPA notice must be authorised by an officer of Superintendent or higher rank, and for ambulance services the Director of Operations must approve the request. Once the notice is authorised it is then processed by a Single Point of Contact (SPoC) within the public authority, who serves the notice to the CSP’s SPoC. As the name suggests, the SPoC is an individual, or a group of individuals, that was/were appointed as the main contact points between the organisations. This ensures that an investigator cannot request any data under RIPA without having the appropriate approval, and that the full process must be followed in order to obtain the data. Thus, the process is self-enforcing, which shows that RIPA and related processes attempt to provide high degree of privacy protection.

Finally, RIPA states that both the notice and the communications data should be transferred in a secure manner according to the DPA, in order to protect the information in transit. However, there is a lack of clear guidelines on how such transfers should occur in the code of practice published by the Home Office [8]. On the other hand, if the investigative data retrieved from a third-party by the police is encrypted, under RIPA, the data subject may be required to disclose the information in an intelligible form or provide encryption key(s) and tools required to render the information intelligible to the investigators [17].

2.3.3 Data retention

Under RIPA and DPA, a CSP should not retain any communications data any longer than it is required for billing purposes, and settling any consequent billing disputes with its customers. Many CSPs do not require the storing of such information for prolonged periods of time, and, consequently, some communication data is disposed of shortly after the monthly bills are issued. However, after 9/11 the Anti-Terrorism, Crime & Security Act 2001 was introduced which allows CSPs to voluntarily retain communications data for periods of time that could allow the UK public authorities to have sufficient information available to them, in order to protect national security [29]. This act did not make CSPs store any more data than required for the billing
purposes, and did not modify any data acquisition procedures stated in RIPA. Instead, the Act simply provided CSPs with an ability to store the data for the periods specified in Appendix A of [29], and maximum of 12 months, without breaching the DPA, RIPA, and Human Rights of the data subject, even when the data is no longer required for business operations. Thus, the individual data controllers could consider storing communication data for prolonged periods of time as necessary in relation to the DPA. This act is often referred to as the Voluntary Code of Practice.

In April 2009, the UK Government issued a public consultation that was the first step of modernising the current approach to data retention [2]. The main reason for this is that many CSPs (especially ISPs) do not require any communications data for billing purposes, anymore. Nowadays, most ISPs charge a monthly subscription fee for the unlimited access to the Internet. Also, it is likely that this will also become the case for telephony providers as they are shifting their operation towards using Voice over Internet Protocol (VoIP) instead of the Public Switched Telephone Network (PSTN). This shift makes the cost of telephone calls negligible, and most calls would not be billed separately but would be included in a monthly fee. Consequently, in the future, CSPs will often have no need to store traffic data for the operational purposes, and the traffic data will no longer be available to the public authorities. In [2] the Government proposes a solution, where all the CSPs would monitor all the Internet transactions taking place over their networks, and would make the traffic data available, and intelligible, to a centralised search engine, referred to as a query hub. Such a search engine could be used by the public authorities in case of RIPA enquiries. If the system proposed by the consultation gets introduced into practice this would modify the rules for data retention and processing towards a more intrusive solution.

RIPA, and other legislations governing communications issues, often refer to the term communications data, which refers to all data about the communications apart of the content of communications. This approach was first introduced with telephone systems in mind. Thus, communication data would refer to: the number dialled and duration of a call (traffic data); the services paid for by the subscriber (services data); and subscriber address (subscriber data); but not the content data, which is the actual conversation, this is the information carried over the telephone circuit during the call.
The division between communications data and content data was relatively easy at the time when all the conversations took place over circuit switched telephone networks. However, the difference between telephone systems and new means of communication, such as Internet, is the fact that traffic data cannot be easily separated from content data. Thus, one can consider HTTP headers as traffic data, since it is used to control the request and response during Web browsing, as an analogy to the telephone service this would be traffic data, since it is used to establish the communication. On the other hand, HTTP headers contain information about the content being viewed, and consequently can allow the investigators to infer a good deal, if not all, of the content data. While most legislations differentiate between these two types of data, there is a lack of clear definitions [30], opening the way for precedent lawyers.

2.3.4 Commission and diligence for recovery of documents

In most of the data-acquisition scenarios considered in this thesis, the evidence is being requested by the public authorities. Other cases, such as when a private party requests data to be provided for a court case, are deliberately left out from the scope of the work as they require a subpoena. However, a particular concept used by the Scottish Court of Session in disputes between private parties finds use in this thesis. Chapter 35 Section 4 of the Rules of the Court of Session specifies that a commissioner may be appointed to fulfil a request for third-party data made by a party in the dispute. This is to ensure that only the relevant evidence are collected from the haver (dataholder) according to the specification of documents prepared by the requesting party, while the data not related to the case, especially haver’s trade secrets, is filtered-out [31, 32]. Similar principles can be found in other legal jurisdictions, for example the discovery escrow in the US intellectual property law [33].

2.4 Privacy, and its wider context

This section provides a brief background on the concept of privacy, and the way that matters of privacy can be examined in an Information System. It is also shown that certain security, auditing and surveillance schemes that were introduced to protect a given population, often breach the privacy of individuals, while other systems with
analogous aims contribute to the privacy of the population that they cover. Therefore, privacy levels in a given information system is not directly dependant on the purpose, but is related to the design decisions and implementation of the given system. A previous section has mentioned that in UK there are plans to increase amount of data that CSPs need to collect in order to enable investigators from public authorities to protect the public and the nation, so this section explains the reasons why some of the proposed measures are justified.

2.4.1 Privacy

A given piece of personal information can be perceived as confidential by one individual, while others would not attempt to conceal it. For this reason it is difficult to define privacy. Stanford Encyclopaedia of Philosophy [34] and [35] provide extensive and neutral discussions on this term. According to these sources some lawyers and philosophers argue that privacy is merely a collection of rights available to an individual to protect the information considered as confidential, and as such, does not merit to be legislated on its own. William Parent defends a view of privacy in the domain of personal information that does not confuse the basic meanings of other fundamental terms. He defines privacy as the condition of not having undocumented personal information known or possessed by others, but he also stresses that privacy is a moral value, and not a legal right.

Perhaps, the best definition for use in this thesis is the slightly wider definition by Alan Westin. This definition describes privacy as the ability to determine for ourselves when, how, and to what extent information about us is communicated to others (as discussed in [34]). Or as Swire and Steinfeld put it privacy is providing individuals some level of information and control regarding the uses and disclosures of their personal information [1]. This definition of privacy may look ambiguous, however, the best practice of handling private data is to allow the data subjects to have a certain amount of control, as no individual is the same as another. Nevertheless, for legislative reasons, there is a need to expand this definition and create laws that could ensure privacy is being maintained by organisations that have access to personal information. In the previous sections the DPA has been discussed as the UK legislation defined for this purpose, and the following section outlines the origins of the DPA, and different views on privacy by different social groups.
2.4.2 Measuring privacy

One of the first comprehensive guidelines for creation of privacy laws was the US Code of Fair Information Practices, developed in 1973 by US Department of Health, Education and Welfare [36]. The document describes five key factors required to achieve privacy:

- **Openness.** Data subjects should be aware that a system keeping their personal data exists.

- **Disclosure.** There must be a mechanism for the data subject to access their own records, and to find out how these records are used.

- **Secondary usage.** Gathered data can only be used for the purpose it has been collected for, unless there is consent from the data subject to further process the data.

- **Record correction.** If the records are incorrect, the data subject should have the right to request a correction.

- **Security.** Where the reliability of the records for their intended use is required, the data controller must take precautions to prevent misuse of the data.

These, and other guidelines, were later adapted by the Organization of Economic Cooperation and Development (OECD) to form the Guidelines on the Protection of Privacy and Transborder Flows of Personal Data. The OECD is formed by 24 countries, where the creation of the guidelines was to harmonise the efforts of the member countries in creating privacy laws [36]. This, in turn, greatly influenced the European Data Protection Directive and, finally, the UK DPA. Consequently, the OECD guidelines, and the DPA, show the way for the system designers on the ways to ensure privacy of the users. However, this is a bare minimum that a system must meet.

The DPA, like most guidelines before it, allows any kind of data collection and processing, as long as the user gives consent to such operation. However, the consent is often only implied, from the fact that the data subject uses a given service offered by an organisation. What is more is that an individual wanting to use a services of a
communication service provider in the UK often must agree to the terms of use, that likely include references to information interception. This also applies to other operations where consent is required. For example, most often banks have similar privacy policies, and for a user to be able to have a bank account, such a policy must be accepted. Consequently, it can be argued that an individual does not often have a choice, and therefore the requirement of consent does not necessarily improve privacy. This confirms claims of [37] that with the evolution of new technologies, the sets of rules proposed in US Code of Fair Information Practices has become outdated. In [37], Marx proposes 29 questions that may be used to help assess the ethics of a given surveillance process. In order to define the framework, Marx identified conditions which, when breached, could violate an individual’s rational perception of privacy and dignity. These conditions call for the following in any surveillance system:

- avoiding harm;
- ensuring validity of reasons;
- building trust with the data-subjects;
- giving notice;
- obtaining permission when crossing personal borders.

Marx’s framework is not technology specific, in order to keep the framework universal and lasting longer than the ever-changing technologies of surveillance. March explains such approach in the following words: *in matters so complex and varied we are better served by an imperfect compass than a detailed map* (Marx, [37], pp. 17).

2.4.3 Privacy in surveillance systems

Western society is subject to many forms of surveillance on a daily basis. Some surveillance activities are overt such as CCTV monitoring, while other are hidden beneath a cloak of customer rewards schemes, or they are being completely concealed from the public eye. One of the surveillance projects that for many years has been concealed from the public was Echelon, which is international surveillance
operation monitoring international communications links [38]. Also, the behavioural advertising scheme by British Telecom, the Phorm, has been kept secret for many months, despite consisting of technology that analyses content of the Internet communications [39]. The UK Government did not react to the breaches of privacy by Phorm, as the scheme formed a useful and readily accessible under RIPA source of surveillance information for the public authorities. This is because, according to RIPA, the public authorities cannot require the ISPs to collect content data for all Internet communications. However, a different situation arises when the ISPs has logs of the content data, as well as a list of interests for every user, collected for business purposes. In such a scenario an investigator could request such data without a warrant. It appears that a significant percentage of monitoring activities are performed in an unethical manner [40]. Often such the unethical surveillance systems are created for the greater good of society (e.g. [41]), and whilst this is mostly true taking into consideration their aims, sometimes the ends do not justify the means, and society often does not benefit from these monitoring systems. In [42], researchers reported on the effectiveness of a number of CCTV deployments in UK. Their findings showed that only a few of the systems achieved their goal of reducing crime levels. The reason for this was that despite the reduction of crime being the initial objective, the design stage of the deployments was not bound to achieving this objective, neither were the way the systems were managed.

In the digital world, the surveillance measures, equivalent to physical CCTV, are logging, monitoring and auditing. Interestingly, in [1] Swire and Steinfeld showed that the implementation of surveillance measures in information systems does not have to go along with lowering privacy of the users. They argue that auditing and monitoring of information systems are standard procedures, and without these, the privacy of data, could be in greater danger, since in an unprotected system the data could be easily stolen or misused [1]. Even though it easy to understand the worries of the general public where their privacy is concerned [43], well-designed and configured surveillance systems may actually stop breaches of privacy from occurring. Swire and Steinfeld believe that security and privacy are complementary, as there are common goals between the two terms. They both are concerned with stopping unauthorised access, use and disclosure of personal information [1].
However, in order to protect the security and privacy of the data subjects, any surveillance system should adhere to firm privacy rules.

### 2.5 Cryptography

Cryptography is the science of keeping data secure from eavesdroppers during transit. Data in its unsecured form is often referred to as plaintext, or cleartext. One method of securing data is encryption and the encrypted data is referred to as ciphertext. A ciphertext may be transformed back into the original plaintext by decryption. When discussing cryptography a set of common symbols is used to donate these operations and states of data. In this thesis the following symbols are used:

- \( M \) – plaintext.
- \( C \) – ciphertext.
- \( E \) – encryption operation.
- \( D \) – decryption operation.

Using these symbols, cryptographic operations can be written in an algebraic form. Thus, encryption \( E \) of a plaintext \( M \), that produces a ciphertext \( C \), is shown in Eqn. 2-1. Similar equations may be used to describe decryption, and other cryptographic operations.

\[
C = E(M)
\]

**Eqn. 2-1**

In the past many cryptographic algorithms, also referred to as ciphers, were based on the secrecy of the mathematical functions that were used to encrypt and decrypt messages. This solution did not scale well, as the mass use of any single cipher was impossible, and every group requiring to communicate securely would need to develop a new mathematical function that could not be broken by the eavesdroppers. Nowadays, only the algorithms that are open to scrutiny of the public and cryptanalysts are perceived as secure [44]. In these algorithms, the secret is protected by the secrecy of the key used during the encryption process, rather than the secrecy of the cipher, itself. In arithmetical notation the key is donated by \( K \):
This section describes the different types of cryptographic protocols, together with their advantages and disadvantages. In cases where the specific source is not provided, the information is based on Schneier’s comprehensive reference book [44].

2.5.1 Classification of cryptographic protocols

Traditionally, most open cryptographic algorithms used the same key for both encryption and decryption, or one key could be simply derived from the other. These algorithms are referred to as symmetric algorithms, or secret-key algorithms, since, in the case of such ciphers, the key needs to remain secret to anybody outside the trusted domain. Consequently, in order for a number of parties to exchange secret messages they had to first exchange the encryption key. Some methods of dealing of this problem included out-of-bound communication, or calculation of a common key by two remote parties based on the Diffie-Hellman (DH) protocol [45] in a way that an eavesdropper cannot produce a valid secret key. The first solution was practical only for the parties that knew in advance that there will be a need to exchange information securely, and that then could use out-of-band communication means (such as secure post) to exchange cryptographic keys. The second, enabled by the DH protocol, lacked means of authenticating the remote party, and although the session key for data exchange could be securely communicated between two parties using DH, the cryptographic techniques known at the time did not allow the parties to verify the identity of each other.

In the same document as the revolutionary protocol of DH was first published, Diffie and Hellman discussed a mechanism that would allow an encryption key to be published to the world without jeopardising the secrecy of any message encrypted under such a key [45]. These resulted in a number of protocols that allowed for asymmetric public-key cryptography to surface [46] including the Rivest-Shamir-Adleman (RSA) protocol [47]. (Mechanism of the RSA for practical use is described in [48].) Such protocols mitigated the need for exchanging the encryption keys using out-of-band techniques, since different keys were used for encryption and decryption, and they were difficult to be derived from each other by anyone else than the creator of the key. These protocols were classified as asymmetric as the
The main use of the private and public key algorithms is encryption. Private-key algorithms are usually symmetric and, for this reason, are fast and suitable for data encryption. Asymmetric public key algorithms are generally slower and considered as less secure, and therefore they are often employed to encrypt session keys for the private key algorithms. However, as previously mentioned, the public key algorithms can also be used to verify the creator of the message.

Cryptosystems can also be classified based on the level of security they offer. Consequently, some cryptosystems may be considered as information-theoretically secure. This term is derived from the information theory developed in 1949 [49], and in the context of cryptography and security would classify a cryptosystem as secure if the original cleartext message could not be recovered from a given piece of ciphertext by a cryptoanalyst with no access to the appropriate decryption key. A few decades later Shamir and other researchers argued that in practice the important distinction is not between doable and the undoable, but between the easy and the difficult (Shamir, [50], pp. 583). The reason for this was that a number of useful cryptographic protocols were then (and still are) based on mathematical problems that were (and most of them still are) hard to solve. A problem would be considered hard if the solution could be calculated but the time taken for this calculation would make the results unusable. Thus, the cryptosystems where it is hard to derive the original cleartext message from the ciphertext are considered as computationally
secure [50]. To give an example: one-time-pad cryptosystem based on a exclusive-OR (EX-OR) operation between the cleartext message and the key, equal in length to the message, would be classified as information-theoretically secure, whereas the RSA cryptosystem is classified as being computationally secure.

### 2.5.2 Authentication, Integrity and Non-repudiation – Public Key Infrastructure (PKI)

The uses of cryptography are not limited to encryption and decryption of messages, and there are a number of additional functions that are needed to handle secure information processing on computers. Thus, the identification of the message sender can be performed using authentication mechanisms. Integrity checking can detect any changes to the message, after it has been formed by the sender, so that it can be verified that the received message is valid. Finally, thanks to the nonrepudiation mechanism, it can be proven that a given message was originated by a given sender. These functionalities allow the communicating parties to trust in the authenticity of information received, in similar way that signatures, seals and tamperproof envelops used to do it in the physical world. Consequently, any system designed to transfer information between remote parties must be capable of performing such checks. However, these functions do not have to be limited to verifying encrypted communications, as they are also centric to watermarking and copyright protection of digital goods [51-53].

These functions are possible thanks to a mix of the public and private key cryptography, and also cryptographic hash functions. Cryptographic hash function is a deterministic procedure that converts input data into a fixed-length bit string (or array), referred to as a hash signature. Such a hash signature can be used to uniquely identify the data that was used as the input to the hash function since any commonly accepted cryptographic hash function have the following properties:

- A small change in the input data results in a large difference in the output.
- Hard to reverse.
- Hard to find two different sets of input data that produce the same output.
- Easy to compute.
Thanks to these properties, the hash functions can be used to verify correctness of the input data if the hash signature of the valid input is known. The two most commonly used hashing protocols are the 128-bit MD5 and 160-bit SHA-1. MD5 has been found vulnerable to a number of theoretical attacks, with a successful attack published in 2008 [54], but still it is widely used. SHA-1 has also been found to be weaker than initially expected, and a collision, a different input data that produces the same results, can be found in $2^{63}$ operations [55]. For these reasons it is recommended that SHA-2 (or stronger algorithms) is used in the applications that are currently being developed [56].

2.5.3 Operations on encrypted plaintext

In certain cryptographic protocols the mathematical operation on the ciphertext has a regular effect on the plaintext. This property is referred to as homomorphism. For example, multiplication of a ciphertext created with the RSA protocol will result in a multiplication of the plaintext. Thus, if the ciphertext of RSA is multiplied by an encrypted number two, after it is decrypted, the value of the plaintext will be twice the original plaintext.

Another homomorphic cipher is ElGamal that also allows for multiplication of the plaintexts [57, 58]. Some other homomorphic ciphers can perform the addition of encrypted plaintext, such as Paillier [59]. These protocols, have already found use in verifiable electronic voting systems [57], and other applications which require privacy and security. However, only recently, a homomorphic cipher which can perform both addition and multiplication was invented [60]. This cipher has not matured yet, but once it passes the scrutiny of peer review, it should be capable to securely evaluate any function (or circuit) over a ciphertext. Consequently, many novel privacy and security solutions could be based on this cipher, and their scope can only be limited by a poor computational performance of the cipher.

2.5.4 Commutative Cryptography

Many cryptographic applications employ sequential encryption and decryption operations under one or more underlying cryptosystems. The reasons to sequence (cascade) different cryptographic schemes together include: strengthening the resulting ciphertext; and achieving additional functionality, which is impossible
under any given encryption scheme on its own [49, 57]. A basic cascadable cryptosystem can consist of a number of encryption stages, where the output from one stage is treated as the input to another. In such a basic cascadable cryptosystem it is necessary to decrypt in the reverse order of encryption operations. However, a special class of sequential cryptosystems – commutative cryptosystems – allows for the decryption of a ciphertext in an arbitrary order. In conventional cryptography when a plaintext message is encrypted with two different cryptographic functions $E_A$ and $E_B$, the resulting ciphertext will be different depending on the order of the key application (Eqn. 2-3).

$$E_A(E_B(M)) \neq E_B(E_A(M))$$  \hspace{1cm} \text{Eqn. 2-3}$$

For most cryptographic applications, this is a desirable behaviour, as it improves the security of the plaintext and the encryption keys. However, commutative protocols are characterised by the opposite property:

$$E_A(E_B(M)) = E_B(E_A(M))$$  \hspace{1cm} \text{Eqn. 2-4}$$

These protocols are slower to execute than symmetric encryption, but they are almost on par with public-key cryptography algorithms. The commutative encryption protocols, in a similar fashion to the homomorphic encryption protocols, can bring a good deal of benefits to the areas of privacy and secrecy [50, 57]. The property shown in (Eqn. 2-4) makes these protocols an ideal choice for testing inputs for equality without revealing these inputs, which will be expanded upon in Chapter 3.

A typical example of commutative cryptography is the Three-Pass (3Pass) protocol designed to enable two parties to share a secret without exchanging any private or public key. The 3Pass protocol can be described using the following physical analogy:

1. Alice places a secret message in a box and locks it with a padlock.
2. The box is sent to Bob, who adds his padlock to the latch, and sends the box back to Alice.
3. Alice removes her padlock and passes the box back to Bob.

4. Bob removes his padlock, and this enables him to read the message inside the box.

Figure 2-1 Analogy to the operation of the three-pass protocol

A more formal, graphical notation of this protocol is shown in Figure 2-2. Using this protocol Alice and Bob can share a secret without sharing a key first and without using a PKI infrastructure. This protocol is aimed at providing an alternative to public-key encryption and DH-like key negotiation protocols. 3Pass, though, has never been widely used in this way since it is susceptible to man-in-the-middle attacks [61] and is less efficient than RSA, a common choice public-key algorithm [47]. However, related concepts are commonly referred to the information sharing [62] and information retrieval [63] solutions.

Figure 2-2 Three-pass protocol operation

Commutative algorithms include:

- Pohlig-Hellman is commutative for keys based on the common prime $p$ [50]. Since Shamir was the first person to propose using this algorithm in this way it is often referred to as Shamir's commutative algorithm. Also, RSA can be modified to work in a similar manner due to the link between Pohlig-Hellman and RSA protocols.

- Massey-Omura has improved the above algorithm by performing operations in a specific Galois Field GF($2^n$) [64]. This allowed for faster realisation of
the cryptographic operations than in case of Pohlig-Hellman algorithm where the operations are performed on the GF($p$) [65].

- ElGamal [58], can be used to form a semantically secure commutative algorithm [57] with modification of the universal re-encryption of the plaintext under this protocol [66]. Another way to build a commutative algorithm from ElGamal is discussed in [67].

### 2.5.5 Zero-Knowledge Proofs

Zero-Knowledge Proofs (ZKPs) can be used to test the other party’s knowledge of a secret without that party revealing the secret. Otherwise stated ZKP is a proof that yields nothing but its validity [Goldreich, [68], pp. 5]. During a round of a ZKP protocol verifier, is often referred to as Victor repetitively asking the prover, Peggy, the same question about the object of the proof changing some variables. The question needs to be formulated in such a way that Peggy must know the object of the proof, in order to answer them all correctly. A common example used in the description of the ZKP is a cave with one entrance and two tunnels that are joined by a secret passage (illustrated in Figure 2-3). Peggy knows how to find and to open the secret passage, and she wants to prove to Victor that she knows this secret. However, at the same time, she does not want to reveal the secret to Victor. The method that she can use to convince him is illustrated by Figure 2-3 and described in Figure 2-4.

![Figure 2-3 Illustration of Zero-Knowledge-Proofs](image-url)
1. Victor stands at Point A and observes that Peggy is going into the Cave.
2. Peggy walks all the way to the secret passage, choosing a tunnel at random.
3. Once Peggy has disappeared, Victor walks to Point B and specifies the tunnel Peggy should use to get back to Point B.
4. Peggy complies, opens the passage if required and comes out the tunnel specified by Victor.
5. Peggy could have already been in the tunnel that Victor has specified, so there is a chance she did not have to use the secret passage. For this reason the whole process is repeated $n$ times.

Figure 2-4 Simplified Explanation of Zero-Knowledge-Proofs

An interactive ZKP protocol would require at least 10 rounds of the above protocol to prove to Victor that Peggy knows a secret. However, a non-interactive ZKP would require Peggy to perform 64, or even 128, iterations of the cut-and-choose protocol, in order to be treated as a valid proof.

2.5.6 Cryptanalysis

Cryptanalysis is the science of breaking the security of cryptographic ciphers in order to decrypt a specific ciphertext, or to obtain a cryptographic key used for a certain purpose. According to Swenson in [69] there are a few forms of possible attacks on any given cryptosystem. Ciphertext-Only Attack takes place when the cryptanalyst have access only to the ciphertext, and is looking to decrypt the message, and find the key that was used to encrypt it. Thus, all protocols must withstand this kind of attack as it assumed that the ciphertext can be made, or may become, public. In many systems it is possible for the cryptanalyst to obtain the plaintext associated with a given ciphertext. If this is the case such an attack is referred to as Known-Plaintext Attacks, and the objective is to derive the key that was used to encrypt the messages. A similar form of attack, listed separately by Swenson is the Probable-Plaintext Attack, is where the cryptanalyst has got a fairly good idea what certain parts of the ciphertext contain, which allows for easier deciphering of the message and deriving
the decryption key. For example if the cryptanalyst tries to analyse a piece of a
source code the ciphertext would contain a large amount of text that are a part of the
programming language used. Known-Plaintext Attack help in decoding the Enigma
code during the World War II [70].

Chosen-Plaintext Attack is the most powerful attack. In this attack a special
plaintext, prepared so that its ciphertext could reveal certain information about the
key used, is fed into the encryption process. It is a powerful type of an attack;
however, a careful design, implementation and exploitation of cryptographic
technologies should prevent a possibility of such attack. A variation of this attack,
suggested by Schneier in [44], is the Adaptive-Chosen-Plaintext attack, where the
cryptographer may modify the plaintexts during a chosen-plaintext attack based on
the results obtained in the previous attack. Finally, if the cryptanalyst have access to
the decryption mechanism, i.e. in form of a decryption process on a computer, or is
capable to eavesdrop a plaintext obtained from ciphertext submitted to a given
process, a Chosen-Ciphertext Attack may be rolled-out with an aim of obtaining the
decryption key, or reverse engineering the decryption process.

As an addition to this list [44] provides two forms of attack not discussed earlier.
Chosen-Key Attack is an unusual attack, in which the cryptanalyst has some
understanding of the relationship between different keys. In [44] Schneier discusses a
chosen-key attack against modification of Data Encryption Standard (DES) protocol,
where the key is rotated two bits after every round. Such attack proved to be
efficient; however, impractical since the variability in rotations of the key in DES
mitigates any chances of rolling-out this kind of an attack on this protocol. Rubber-
hose cryptanalysis (or Purchase-Key Attack) is often ignored by the system
designers, but it is one of the most powerful attacks. It is a form of social engineering
attack, where the cryptanalyst forces, or bribes, someone to deliver the key.

The classification of attacks on systems using cryptography alone shows the
significance of the system design and implementation. The first form of attack can be
rolled out against virtually any system, however, the other attacks are often a result
of implementation error or procedural error in the way the protocol is being used
[70].
2.5.7 Compliance

Public organisations, depending on their type, connect to different government computer networks, such as Government Secure Intranet (GSi) network [71]. This way the data can be kept relatively secure within the parameter of the network. Dependant of to which network they connect a compliance to a different standard may be required for the methods of exchanging sensitive data with third parties.

In the UK, the Communications-Electronics Security Group (CESG) is the national technical authority for information assurance. CESG Assisted Products Scheme (CAPS) examines and lists cryptographic products that can be used to protect data in transit any of the classification levels [72]. There are three levels of security that a security product can offer according to CAPS:

- **Baseline.** Products meeting this compliance level can be used to protect information up to and including RESTRICTED classification. PGP Whole Disk Encryption produced by Symantec falls into this category.

- **Enhanced.** This is required to protect CONFIDENTIAL information. BeCrypt DISK Protect Enhanced is an example of such product.

- **High.** Products that meet requirements of the high protection under CAPS can protect information of SECRET or higher classification.

Another scheme for assessing the level of protection given by the security product is the Common Criteria scheme committed to by a number of different countries – including the UK, USA, Canada, Germany, France, New Zealand, Australia, and others. This allows for all the participating countries to develop common requirements for standards and to evaluate products against those standards [73].

Finally, when the requirements are less strict or the CAPS or Common Criteria products can not be used in the given system, then CESG Claims Tested Mark (CCTM) can be used to verify that the products performs to the advertised standards [72]. In this case, it would also be beneficial to verify the product against the Federal Information Processing Standard (FIPS) 140-2 [74] that specifies the adequate algorithms and their implementation that are required to handle PROTECTED documents (or CONFIDENTIAL in risk managed environment).
2.5.8 Trusted Computing (TC)

TC is commonly perceived as a technology that protects the interest of large organisations if their software is executed or data processed on end-user devices. Consequently, TC could be considered to invade privacy and limit freedoms of individuals. However, this is not the case, as TC is simply a technology that can provide secure encryption, authentication and attestation services [75]. It is thanks to these services that software manufacturers can make sure that their applications can be installed, only on licensed devices and that data providers can encrypt software so it can be only read by a specific application running on a specific licensed device. This limits freedom that an end-user had in regards to use of software and data (such as music files, videos and documents) before TC came into play. Nevertheless, TC can also be used to protect the users, by encrypting their data and providing secure authentication services to remote servers. Going further, the scenario in where rights of users are being limited can be reversed by placing TC base at the server-side of the services in order to provide users with enhanced privacy and reassurance on how information about them is handled [76]. Thus, the users could execute different procedures on the server with the service provider knowing limited detail of their requests [75].

Trusted Computing Group (TCG) defined a standard for Trusted System (TS), which defined a technique called memory curtaining that restricts the operating system from accessing certain areas of the system memory. However, this does not give a great deal of protection against a perpetrator with access to the hardware. Such perpetrators could simply monitor the memory paths between the motherboard, storage devices and RAM, in order to obtain intelligence on actions initiated by the user [77]. Thus, the solutions proposed by the TCG are not designed to be fully tamperproof. Nevertheless, it is possible to employ Secure Coprocessor (SCOP), a computing environment, co-located at a host machine, which can perform security-sensitive computations to provide the following assurance [78]:

- Bits encrypted with a given application’s public key can only be decrypted by this application running on untampered TC component.
- Bits signed by this application’s private key were produced by the application running on untampered TC component.
SCOPs are typically built with general-purpose computer hardware, with addition of hardware cryptography accelerator and tamper detection/reaction mechanism. A typical example of SCOP is IBM 4758 platform that connects to a PCI slot in the host machine. FIPS 140-1 Level 4 validation of this device is a proof of its sound design.

2.6 Conclusion

In UK there are two major legislations that regulate collection, retention and release of personal data: DPA and RIPA. DPA regulates the processing of personal data that is stored electronically, or in an organised fashion that allows for retrieval of data about a specific individual. This legislation aims to guarantee lawful and fair processing of the data. Under DPA the data controller can voluntarily release personal data if required by certain public authorities. The provision can be used by police investigators, and the intelligence services, as well as accident and emergency staff in health care.

RIPA has been introduced to regulate investigatory powers that were frequently abused. It permits CSPs to collect and retain data, other than transaction content data, if it is needed for the system maintenance or billing. Under RIPA the public authorities have the right to request data from CSPs without subpoena or any justification, and the data controller must provide the requested information or face penalties. Thus, RIPA, in comparison to DPA, gives investigators better access to data.

Once always-on Internet connections started to emerge, telecommunication companies introduced all-inclusive call tariffs, and the logging of traffic data was no longer required. Therefore, CSPs had no longer an incentive, nor legal right, to retain this data. This could cause UK investigators to loose one of the commonly-used investigative data sources, and the UK government introduced the Voluntary Code of Practice that allowed CSPs to retain this data for up to a year without breaching the DPA. Later, in April 2009 another modification to the laws governing the personal data was proposed by the Government. The proposal suggested the necessity to create a distributed database for all the traffic information from the UK-based CSPs, including data in collected from transit connections. Consequently, the consultation
document confirmed that the assumptions made during formulating the aims of this thesis aims were right, and the UK Government is looking into introducing new more intrusive legislation to replace and/or extend RIPA, and to widen the investigative rights.

This chapter also discussed the concept of privacy. Unfortunately, there is no clear definition of privacy, and some argue that, in legal terms, privacy is just a collection of rights, and not a legal term, while others recognise privacy as a moral value, and not a legal right. However, most definitions have a common factor, where there is a notion of control that an individual should have over the way their personal data is used. One of the first instances of documented privacy guidelines was the US Code of Fair Information Practices developed in 1973 by the US Department of Health. This code identified five key areas in fair information processing: openness; disclosure; secondary usage; record correction; and security. Despite the code being prepared more than 30 years ago, the legislations in the UK and Europe, including the DPA, widely inherit from it. However, the code of practice, and the DPA itself, are not specific enough to assist system designers in the building and evaluating systems that respect privacy. The DPA can thus merely assess whether the user is provided with an appropriate level of control over their own personal data. For this reason another technique of assessing privacy in information systems is necessary. In the case of surveillance systems, some researchers proposed a series of questions that can be used to verify the aim, and the means of the system in respect to privacy.

Any process that is used to collect data that may required to be presented in front of court of law should follow evidence handling guidelines. The guidelines respected by the UK public authorities, and especially police operations, is the Good Practice Guide for Computer based Electronic Evidence put together by the ACPO. There are four principles presented in this guide: no action of the process should alter the evidence; when the evidence needs to be accessed directly or destructively tested such operation needs to be performed by qualified person that can later give evidence in court; strict audit trail needs to be maintained; and the law and these principles must be adhered to at any time when handling the data.

Finally, the chapter provided background to cryptography and basic cryptographic techniques. Thus, the symmetric algorithms that employ the same key for encryption
and decryption operations, and asymmetric algorithm that use different keys for these operations were compared. Then the distinction between faster, more secure symmetric cryptography and the versatile asymmetric cryptography that enabled for existence of ecommerce and modern secure communications were revealed. This chapter also detailed different types of attacks that could be employed against a cryptographic protocol.
Chapter 3

Literature Review

3.1 Introduction

This chapter reviews the research related to the most important concepts in the field of privacy-preserving investigative data acquisition. Section 3.2 identifies and discusses the research closely tied with the subject of this thesis, while Section 3.3 provides an insight into the primitives referred to by the research presented in Section 3.2.

The literature related to technologies and techniques allowing individuals to protect their privacy and control the data that relates to them is discussed. It is shown that despite the relevant privacy legislation, mainly the DPA, giving the consumers an option to opt-out from being a data subject in a system; financial and convenience factors may force the consumers to opt-in and use a system that they consider as privacy intrusive. Therefore, individuals that wish to preserve their privacy reach for PETs. Anonymous Internet browsing can be achieved by employing onion routing
techniques, while the identity of a buyer (and, in some circumstances, the type of goods purchased) can be hidden from the seller with PIR and private comparison protocols.

The current privacy preserving measures during investigations conducted by the public authorities are shown to be limited to policy-based controls. As discussed in this thesis, these controls are not capable of hiding the identity of the data subject being investigated. Related research shows that it is possible to create surveillance systems with a set of hidden criteria that triggers and alerts if a monitored individual performs a forbidden action. It is also possible to create a privacy-preserved blacklist that informs the investigators about transactions performed by listed suspects.

Data collected during investigations needs to be stored with the same controls as digital evidence. It is also important to ensure that the source of the data is valid and has not been tempered with. For this reason literature relating to preserving of digital evidence is also reviewed in this chapter. Techniques for minimising amount of storage required for the evidence, and allowing data to be verified for authenticity are discussed.

3.2 Privacy-Respecting Investigative Data Acquisition

Chapter 2 introduced the investigative data acquisition process in UK. This section provides information on on-going research in this field.

3.2.1 Privacy self-defence

Research suggest that individuals have different attitudes to privacy depending on the service that they use, and depending on the organisation that is collecting the data [79]. Surprisingly, some individuals that report high levels of concern about privacy, do not consider giving away their personal data in exchange for services and goods as a serious privacy issue [79, 80]. As discussed in Section 2.4 privacy is a complex matter, thus, privacy decisions should be left to the individuals they concern. Thus, the consent of the data subject is important to the privacy guidelines [36] and legislations [7, 10]. This would suggest that the best privacy defence measure available to individuals would be an opt-out mechanism that all organisations must provide under the relevant legislations, it should be noted that even after opting-out,
some organisations, and, especially public institutions, will retain the right to keep certain private data indefinitely [10]. However, research shows that there are situations where the user must opt-in and cannot opt-out. Such situations are usually enforced by convenience and economical, rather than legal factors [23]. An example is the prepaid travel card for London public transport – the Oyster card, where the movements of a person paying with the card can be easily tracked, and as [23] states, the data collected by the back-end of the Oyster system is already being used by investigators from public authorities. Although it is possible to travel around London using cash, paying with the card works-out cheaper than with cash. There are ways to purchase this top-up card without registration, which requires paying a small deposit fee [81]. While this is inconvenient, once the serial number of the Oyster card belonging to an individual is identified using another means, such as CCTV footage from the time it has been topped-up or purchased then the movements of the data subject can be traced-back in the system. Similar considerations apply to access to mobile networks and Internet, as well as other services desired by consumers. Consequently, choosing not to opt-in (or choosing to opt-out) is sometimes not an option and, for this reason, individuals looking to have their privacy protected need to use other means to achieve this.

Onion routing networks, such as The Onion Router (TOR) allow for anonymous Internet browsing. This is an interesting approach for privacy that conscious users can use to protect their identity. The infrastructure is based on a series of relays that TOR clients can use to route their requests. These relays are simply network nodes belonging to clients that allow other clients to use their bandwidth in order to create a coherent anonymising network. Since the traffic between the client and relays within the network is encrypted, any request made through TOR to an Internet server can only be traced-back to a relay that executed the request, and not to any particular client. Even the participating nodes are unable to trace-back the request, since the requests are re-encrypted at each relay in the path. Thus, the IP address of a requestor is safely hidden in the population of active TOR clients [82]. However, there are transactions that require at least some form of authorization of the participants. As an example, a website may require all the visitors to be of a legal age, in order to enter. Microsoft researchers suggest that Identity Metasystem, that manages privacy during authentication and authorisation, could be the answer to this concern. The Identity
Metasystem can be based on the data minimisation principal (also discussed in Section 3.3.1) and limit the information released to the minimum required to perform a transaction. Such a system would work as a middleware for all identity-related interactions [83].

Some solutions from the area of Multi-Party Computation (MPC) could prove to be more effective in protecting the privacy of the Internet transactions (and in physical authentication as well). Thus, as an example, systems allowing for anonymous digital payments exist in enabling privacy-preserving purchases of electronic goods and services. The digital money, or *ecash*, has been designed in a way that makes the buyer untraceable, as long as there is no fraudulent attempt to reuse the *ecash* in another transaction. This ensures that the customers can make anonymous transactions, as long as they do not try to cheat the system [84]. Unfortunately, the *ecash* does not address the fact that the seller will know which goods have been purchased, and when. This can potentially help the seller to profile an anonymous buyer or inference some information about the buyer’s identity, and thus, has been addressed by the protocols described in [85]. It is important to note that stopping the seller from finding out this information can possibly stop the seller from optimising sales. Consequently, the privacy of the buyer is protected at a cost to the seller. However, the buyers can then voluntarily provide some information, or feedback, to the seller. This approach is *in-synch* with the true spirit of privacy, where the individual described by the data is in control of the data.

### 3.2.2 Privacy Controls in an Investigative Scenario

Technology-based solutions to protecting privacy of potential suspects are not commonly used in practice. The most widely deployed controls in this area are processes enforced by data protection and human rights legislations. In [8] a code of practice for using the investigatory powers granted by RIPA is provided. It specifies the process for granting the authorisations and giving data acquisition notices, that involves four different roles:

- **Applicant.** A person that requests the data needed for a specific investigation or operation within a relevant public authority.
• **Designated Person.** An individual responsible for assessing the application for data acquisition that ensures the request is *necessary and proportionate*.

• **SPoC.** This could be an individual or a group of accredited individuals trained in lawful acquisition of communications data and co-operating with third parties.

• **Senior Responsible Officer.** The officer oversees the whole data acquisition process to ensure compliance with the appropriate legislations.

Out of these four roles, the designated person is delegated to protecting the rights of the data subjects. Consequently, the whole process is organised to ensure that the data acquisition request is *necessary and proportionate*. The key term here is *proportionate*, as it states that the amount of the potential collateral damage caused to the data subject is justified by the objectives of the request. Furthermore, as described in Chapter 1 the collateral damage can (and is likely to) occur, since in order to obtain data about an individual, the identity of this individual must be revealed. This is the case if only process-based privacy controls are used to safeguard privacy. However, the research discussed in [86] suggests that there are privacy-preserving primitives that can be used to provide greater privacy levels in the investigative scenarios. The suggested primitives are:

• *asymmetric equality* allowing two parties to compare their inputs without revealing them;

• *split comparison* providing the participants with the ability to compare inputs, once again keeping them secret from each-other;

• *equality with selection* that can be used to provide the requestor with a different output depending on the result of an *asymmetric equality* test.

These primitives are combined in [86] to form a system for privacy-preserving electronic surveillance, that allows the tracking of electronic transactions performed by individuals listed as potential suspects. This size of the set (referred to as *n*) containing identities of the suspected individuals is assumed to be much smaller than the population (donated by *N*). The schemes described in [86] assume that it is tractable to perform O(*N*) operations (270 million, the population of United States, is
the number used), but it is also noted that performing this many operations is impractical. Thus, trading suspect privacy for speed is discussed [86]. Each asymmetric equality test performed to check whether a given identity belongs to one of the $n$ suspects would typically use two unique sets of public keys. However, as [86] suggests, if these keys are reused in more comparison rounds, the communication cost, as well as the number of computations required, are greatly reduced. This is also the technique suggested in [63] where a system for selective sharing of information between parties is discussed. The main drawback of this technique is providing the potential perpetrator greater scope for a known-plaintext attack. Also, once the key is broken all, identities encrypted by this key are revealed.

Another approach to ensure that investigators or auditors can only review actions taken by certain individuals already considered as suspects, is proposed by Biskup and Flegel in [87, 88]. Their system can be used as a pseudonym-based privacy-preserving middleware for audit software. The identities of the users are hidden using pseudonyms unique per identity, or per transaction. While the first case allows auditors to identify suspicious activity of an individual hidden by a pseudonym, the second case makes it impossible to study (or profile) actions performed by individuals, thus providing full anonymity to the users. Auditors can however reveal identities of the pseudonyms after a warrant is given for the given pseudonym or individual, while the system reacts to the transactions taking place by automatically revealing identities of individuals that perform a number of prohibited transactions. The system is based on Shamir’s approach to splitting a secret based on polynomial interpolation, so that the core of the system is not based on cryptography. Still, these systems are impractical for large-scale implementations, where it is not feasible to associate each transaction with a unique and tractable pseudonym. Certainly, such a system could not exist in scenario involving a population in the region of $N=270$ million, as proposed in [86]. As Kantarcioglu and Clifton describe it, privacy is not free [89] and keeping private information secret requires many computations and many communication rounds.

Researchers note that investigators can easily trick systems into providing data about innocent individuals by simply placing these individuals on the privacy-preserved lists of potential suspects. The solutions proposed in [86] and [90] suggest that a
warrant signature system is needed, where the party in-charge of data acquisition warrants signs a request to assure the participants that the investigation is authorised.

An interesting problem is considered in [89], where the government wants to split users of a certain third party system based on secret classification criteria, while the privacy and equality advocates want to ensure that the criteria is fair and that no data about the users is provided to the government agencies apart of the results of the classification. These requirements can be considered as contradictory. However, [89] shows that it is possible to achieve such a system by the use of commutative and asymmetric cryptography, and one-time pads. This illustrates the power of sequencing different encryption schemes to achieve a functionality that would be impossible to achieve using a single encryption scheme alone, as discussed in Section 2.5.4.

### 3.3 Privacy-Preserving Primitives

#### 3.3.1 Privacy-Enhancing Technologies

PET is the common name for *a range of different technologies to protect sensitive personal data within information systems* (Koorn, [91], pp. 1). Such technologies find use in all types of information systems. The discussion in [91] deals with typical scenarios where the owner of the data has an incentive, such as a required legislative compliance, to provide privacy to the data subjects.

There are a number of conventional privacy controls that can satisfy the current legislative requirements, and although these are not as exciting to the academic community, they are still valid solutions to the privacy concerns. Overall, they have been tried and tested over the years, and for this reason they are often compliant with the security standards that dictates data processing in many organisations. In [91] the following types of conventional PETs have been identified:

- **General PET controls.** These are the controls that can be implemented with technologies similar to those used in data security. For this, the privacy is treated as the highest level of security, and in this way is a well-defined security policy where its controls protect the privacy. This type is PETs is further split into:
- **Data Minimisation.** Analogically to the principal of least privilege in security, a minimal access to data should be given to any requestors. If the requestor needs to know whether a given data subject is an adult, yes or no answers would suffice to fully answer this question, without the need to provide the requestor with neither the age of the data subject nor the date of birth.

- **Authentication and Authorisation.** These should really be treated as prerequisites for any system that carries data that is not publicly available. Without these, other controls, such as the data minimisation mentioned cannot be deployed.

- **Quality-Enhancing Technology.** Part of the requirements for fair information processing [36] and DPA is ensuring the correctness of the data. This can be done by improving the data collection mechanism, as well as allowing the data subjects to view and correct the data about them.

- **Separation of data.** This control splits a data-source into two or more domains, where the personal data that carries information such as a name and an address is stored in the identity domain, and the other personal data is stored in another domain against the pseudo-identity that was derived from the real identity. These domains are linked by identity protector software that enables only privileged users to restore the relationship between the data-records in the different domains. Consequently, with this control applied, the personal data can be analysed without revealing the identity of the data subject to the analyst.

- **Privacy management systems.** This type of controls is the least mature of the conventional methods presented in this section. It introduces software that ensures automated enforcement of the privacy policy. Such software intercepts any transactions that involve personal data, and tests these against the privacy regulations, which might include the privacy policy and privacy preferences of the data subjects that the transaction concerns.

- **Anonymisation.** This is a similar approach to the separation of data, where the difference is that the pseudo-identities cannot be linked back to the real identities of the data subjects, nor can the identity of the data subject be
inferred from the anonymised data. Thus, the process of anonymisation transforms personal data into data that can be freely processed without privacy controls.

Figure 3-1 illustrates the effectiveness of these PET types as described in [91]. It is worth noting that, despite the anonymisation techniques being valued the most, these cannot be applied in systems where processing of personal data is necessary, unless the personal data is only processed on the input to the system, and it is automatically anonymised on writes to the database.

At the time of its publication [91] was an authoritative guide to PETs for decision makers it fails to mention PETs such as mixnets [57, 66], crowds [92] and PIR [93]. Thus, there is another type of PETs, which this thesis defines as:

- **Identity hiding.** Whether it is hiding the identity of the interesting record being retrieved from a database in a larger group of records (PIR) or hiding the identities of individuals that committed an action by a group of individuals (crowds and mixnets) it is possible to provide an additional level of privacy by hiding the target or an originator of an action in larger group. Such solutions are usually computationally-expensive, thus, it is more likely to be utilised by individuals wanting to improve their privacy, rather than organisation seeking to protect their data subjects.

In relation to the PET Staircase presented in Figure 3-1 Identity Hiding techniques would most likely perform on par with a Anonymisation techniques in terms of effectiveness in privacy protection. Since, the privacy-preserving solution to investigative data acquisition is unlikely to be found in the classical PET technologies described in [91], as these would be in use by now, the reminder of this chapter focuses on the Identity Hiding techniques that can be used in this domain.

When discussing PETs and privacy-preserving operations on data, it is important to note the distinction between Private Data-Mining and PIR. Both are well researched subjects, however, the first term – private data-mining – is usually used in relation to obtaining anonymised, statistical data rather than retrieval of individual records as it is the case with PIR. While some techniques used in various available approaches to
private data-mining can be modified and reused in information retrieval and vice-versa, these two primitives are dissimilar in objectives.

Figure 3-1 PET Staircase: the effectiveness of the different conventional PET types

In the field of statistical data-mining, researchers have developed a number of techniques that permit operations on a subset, or cross-section, of datasets [94-96]. In [96], Agrawal and Srikant suggest a technique based on perturbations, where the larger the perturbations, the greater the level of privacy in the system, but such a technique can result in a loss of information. However, Agrawal and Yu ([95]) show that this is a natural trade-off between accuracy and privacy, similar to these caused by adding noise to data that is then approximately removed from the output [94]. An interesting approach is based on $k$-anonymity models [97] that ensure that any attempts to link a given record to the data subjects its describes result in at least $k$ different identities being returned. Thus, contrary to the security where any leak of information may be unacceptable [93], privacy can be achieved by hiding the data subject in a larger group of individuals. Finally, a system that does not lose precision can be achieved by employing primitives from the area of MPC, but, in order for such schemes to be feasible, they often need to make use of an extra party in the protocol – semi-honest party trusted not to collude with other participants – otherwise the computational complexity of the protocol may be too high [94].
3.3.2 Multi-Party Computation

MPC allow a number of parties to engage in a protocol that enables them to compare their secret inputs, or to compute a function, without revealing these inputs. It used in scenarios where no trusted third party exists. Therefore, it can be used to solve a function, such as $f(a,b)$, where $a$ is Alice’s input data and $b$ is Bob’s input data without the need to revel these inputs [86, 98]. A classic case is Yao’s millionaires’ problem, where two millionaires seek to compare their fortunes without revealing the exact figures involved [98]. Yao provides three different solutions to the problem, giving the basis for the multiparty computation of two different parties. He also specifies a method to scale up his techniques in order to allow computation for $n$ different parties. However, schemes designed specifically to handle computation between $n$ different parties were later introduced by Goldreich, Micali and Wigderson [99].

MPC protocols are largely based on the same functions as common encryption schemes, and therefore most have strong theoretical underpinning [100]. However, it should be noted that the security of any MPC protocol strongly depends on the function that a given protocol is designed to evaluate. Thus, if Alice and Bob engage in a protocol that can evaluate a function $f(a, b) = a \times b$, the party that learns the result can also calculate the input from the second party without breaking the protocol. Consequently, some functions cannot be evaluated privately. For these an MPC protocol can obfuscate the process and hide security vulnerability rather than solve the underlying problem.

Many MPC protocols, though, are characterised by an exponential growth in the computational complexity with linear growth of the number of records to be processed. These protocols are often impractical while working on large datasets, such as those containing ISP data or health records. The protocols that are characterised by a linear increase in processing time as a response to increased number of records or their size are referred to as efficient protocols.

3.3.3 Sharing a secret

An interesting primitive that is commonly used in the field of privacy-preserving information retrieval deals with sharing a secret with a number of parties. In this kind
of information sharing schemes, a party (Alice) wants to share a secret with another
party (Bob), only if the board of trustees agrees that Bob should have access to the
secret [62]. An alternative description of the problem is that a number of parties want
to lock their secret, so that it can be retrieved only when they co-operate [101]. For
performance reasons, the secret is usually a small piece of data, however, it can be
used to store a secret key to a larger dataset, and thus, such schemes can be used to
pass control of any asset from Alice to Bob. Khayat in [62] employs the 3Pass
primitive described in Section 2.5.4, into a secret sharing with a board of trustees
scheme.

Figure 3-3 illustrates the encryption operation of this scheme, with the description
given in Figure 3-2, while the decryption is done in an analogical way (see Figure
3-4 and Figure 3-5). A trustee may leave the scheme by removing his encryption
from the ciphertext held by Bob. It is also possible for a new trustee to be added to
the scheme, if, for example another trustee is leaving the scheme and a new trustee
needs to be appointed. Unfortunately, Khayat has overlooked issues that could arise
from an implementation of this secret sharing scheme in real-life scenario, such as:

- Death of a trustee.
- Betrayal.
- Corruption of the ciphertext.

**Sharing Secret with a board of trustees – Locking Phase:**

1. Alice encrypts the message using her key and sends the
   produced ciphertext to a trustee.
2. The trustee encrypts the ciphertext received with his
   key and forwards the result to another trustee or to
   Alice.
3. Once Alice receives the encrypted/locked message, she
   decrypts the ciphertext using her key in order to remove
   her encryption/lock.
4. Finally, the ciphertext is transferred to Bob.

Figure 3-2 Locking a secret under Khayat’s secret sharing scheme
Objective: locking a secret by a number of trustees.
Alice’s input: secret message $M$; encryption key $E_1$; decryption key $D_A$.
Bob’s input: encryption key $E_2$; decryption key $D_B$.
Trustee’s input: encryption key $E_α$; decryption key $D_α$; for the $n^{th}$ trustee.

Figure 3-3 Illustration of locking a secret under Khayat’s secret sharing scheme

Objective: unlocking a secret by a number of trustees.
Alice’s input: secret message $M$; encryption key $E_1$; decryption key $D_A$.
Bob’s input: encryption key $E_2$; decryption key $D_B$.
Trustee’s input: encryption key $E_α$; decryption key $D_α$; for the $n^{th}$ trustee.

Figure 3-4 Illustration of unlocking a secret under Khayat’s secret sharing scheme
**Sharing Secret with a board of trustees – Unlocking Phase:**

1. Bob encrypts the ciphertext with his key and sends it to any trustee in the scheme.
2. A trustee removes his encryption from the ciphertext and passes it to another trustee, or back to Bob. The order of decryption (lock removal) by trustees is arbitrary.
3. Once all the trustees removed their locks, Bob can decrypt the ciphertext and can obtain the plaintext message.

Figure 3-5 Unlocking a secret under Khayat’s secret sharing scheme

In the event of such issues arising, the above secret sharing scheme would be broken. However, these issues were considered by Shamir in a description of his own secret sharing scheme [101], some 25 years before publication of Khayat’s scheme. In Shamir’s secret sharing scheme, a secret is divided into $n$ pieces, however, only $k$ pieces are needed to use the secret. Thus, up to half of the pieces may be missing (or corrupted) and the operation of protocol would not be affected, and the secret could be retrieved in the most optimal case: $n = 2k - 1$. Consequently, Shamir’s scheme (described in Figure 3-6), based on polynomial interpolation, and not an encryption protocol, provides an efficient and an adaptable solution for secret sharing. Also, this scheme is information theoretically secure, unlike the trapdoor-based solution proposed by Khayat.

Many different secret sharing schemes exist, and a good overview of these is provided by Schneier in [44], however, most popular protocols are a variation of the polynomial interpolation concept. This concept is based on an assumption that the secret, can be written as a number $D$.

Despite Khayat scheme not being suitable for the intended purpose, the concept of locking and unlocking a secret in an arbitrary order by a number of parties forms a useful digital equivalent to a *safety lockout hasp*. Such hasps are used by engineers to lock-out an area or a resource, while work is being carried out. Thus, each engineer places a padlock on the hasp on commissioning a task, and removes it once the work is done. The hasp can only be opened once all the padlocks have been removed,
meaning that all engineers have finished their allocated tasks. This is an important primitive behind a number of information retrieval schemes described later in this chapter.

**Secret sharing scheme based on polynomial interpolation:**

1. Choose a prime number \( p \) that is larger than \( n \) and \( D \).
2. Pick at random \( k-1 \) degree polynomial:
   \[
   q(x) = a_0 + a_1 x + ... + a_{k-1} x^{k-1}
   \]
   where \( a_0 = D \) and other coefficients belong to \([0, p)\). 
3. Compute \( n \) different 2-tuples \((1; D_1),...,(n; D_n)\) so that
   \[
   D_i = q(i) \mod p.
   \]
4. Distribute the 2-tuples among the participating parties.

---

**3.3.4 Retrieving records in a private manner**

Data retrieval is a fundamental operation in computing. Therefore, there is little wonder that PIR is one of the most researched privacy-preserving primitives. Initially, PIR protocols were designed with a basic requirement of acquiring an interesting data record, or just a specific data bit, from a dataholder, the *sender*, in a way that this dataholder is unable to judge which record is of interest to the requestor, the *chooser*. These protocols were not concerned with the secrecy of other records stored in the database, thus in its least optimised state, a PIR could have been achieved by transferring the whole database from the *sender* to the *chooser*, as this would allow the *chooser* to retrieve a record in a private manner. Consequently, the main motivation behind the research in this field is to achieve PIR with a minimal communicational and computational complexity [102, 103].

There is a firm distinction between a single- and a multi-database PIR protocols. It is possible to achieve PIR with information-theoretic privacy by making a number of requests to the database(s) with a distribution that does not allow the dataholder to identify object of the interest. As expected, this operation is more efficient in the multi-database scenario, and requires minimum of \( O(n^{1/2k-1}) \) communication complexity, where \( n \) is the number of records in the set, and \( k \) is the number of
databases where this set can be obtained from. However, PIR protocols with only polynomial time assurance can achieve much smaller communication complexity, by introducing balancers [102]. These employ trapdoor functions that can be used to change the ratio of computation to communication, and thus can be used to minimise the amount of data that needs to be transmitted for a given run of a PIR protocol. Such a solution was also suggested by Naor and Pinkas [104], who emphasise that selection of the trade-off between computational and communicational complexity depends on the specific problem at hand.

In [105], Shundong discusses a retrieval system that uses symmetric cryptography in order to lower the cost of cryptographic operations, as trapdoor functions are about 1,000 times less efficient than symmetric operations. However, this solution still requires the use of some trapdoor operations. Consequently, the proposed solution is analogical to PKI, where data is encrypted using faster symmetric algorithms, and the symmetric keys are then hidden using a trapdoor function, such as these provided by RSA encryption.

A stronger notion than PIR is the Oblivious Transfer (OT) primitive introduced by Rabin [106]. In its original form it allows two parties to engage in a secret sharing protocol that ensures that both parties provide their secret entries without cheating. Normally, if two parties want to exchange their secrets, one of the parties provides false data in return for the secret from the other party, or back away from the protocol once the secret of the other party has been revealed, but before revealing its own secret. Rabin’s OT ensures that the parties learn nothing if one backs out from the protocol before it completes, and that, in case one of the parties cheats in the protocol, this can be proven at the later date. However, this primitive is the mostly widely used in a more sophisticated form that can enable the chooser to select one out of two values, or records, held by the sender in a way that the sender cannot learn which record has been retrieved, and the chooser cannot learn anything about the other record. This extension to the OT primitive is referred to as 1-out-of-2 OT (1-2 OT) or $OT^1_2$, and in [44] Schneier provides its basic protocol to achieve an 1-2 OT (Figure 3-7).
An Oblivious Transfer protocol:

1. The sender generates two sets of public/private keys pairs, and sends all public keys to the chooser.

2. The chooser generates a key with a private key encryption algorithm, such as AES, later called the AES key. The chooser then uses a public key received from the sender in Step 1 to encrypt the AES key and send it to the sender.

3. The sender does not know which public key has been used to encode the AES key, or which record has been selected, thus protecting the privacy of the suspect. The sender can then attempt to decode cipher-text received in Step 2, using all private keys generated in Step 1, whilst preserving the order in which they have been decrypted. In this way two potential AES keys are created. But only one is the proper AES key; the other output is a random set of bits, which cannot be distinguished from an ordinary AES keys.

4. The sender encrypts the two records using appropriate keys decrypted in Step 3. Thus, the first record is encrypted with an AES key decrypted using the first private key generated in Step 1, the second record is encrypted with an AES key decrypted using the second private. Consequently only one record includes data about the suspect. The record is encrypted using the AES key generated by the chooser in Step 2, sent to the sender encrypted by the appropriate public key, and then decrypted using relevant private key. In this way the selected record will be encrypted using the proper AES key, while the other record will be encrypted using by the random string of bits unknown to the chooser.

5. The chooser gets the encrypted records, but using the AES key the chooser is able to decrypt only the selected record. The other record is unreadable to the chooser provided that the false keys generated in Step 3, and used to encrypt these records in Step 4, are not broken.

Figure 3-7 Basic Oblivious Transfer Protocol by Schneier
In the work describing MPC, Yao provided a technique for scaling up any 1-2 OT protocol into a 1-n OT protocol. However, the 1-n OT primitive that allows for the retrieval of a randomly selected record from the dataset of $n$ elements held by the sender, may not be useful apart when playing mental games [61, 99], as Schneier point out in [44]. 1-2 OT is typically based on modular exponentiation, thus, it is resource intensive. Consequently, even though it is possible to derive 1-n OT from 1-2 OT, in practice 1-n OT, designed as such from the ground up are more efficient [104, 107]. This is backed-up by Goldwasser’s proofs that MPC protocols designed for a specific tasks perform better than the general-purpose protocols [100].

Just like in PIR, the fundamental primitive is designed to operate on bits, while, for most proposed uses, OT of strings is more practical. Protocols that can allow for efficient OT of strings make possible transferring control over any digitally controlled, or contained, asset from one party to another, as access keys and passwords can be retrieved [63, 107]. For example, if the sender wants to allow the chooser to privately purchase an electronic book, it can openly publish the full content of the electronic library with each book encrypted under a unique symmetric encryption key, and then once the chooser makes a payment, the parties engage in an 1-n OT allowing the chooser privately select the decryption key for a given book [85, 92]. This is a common approach for transferring control over a resource from one party to another, however, in the digital bookshop scenario; the sender would need to know which book has been purchased in order to charge the chooser the correct amount. Alternatively all books could have equal prices.

Clearly, both approaches are impractical. A solution to this problem is published in [85], where it is suggested that a buyer should make an initial deposit allowing them to obtain a number of goods. The seller would then need to ensure that the balance of the deposit is higher that the value of the goods being purchased. However, the seller should not learn the exact balance of the deposit, but only the result of comparison between the value of the goods being purchased and the deposit balance (a general protocol for making such private comparison is later discussed in Section 3.3.5). Consequently, the balance of the deposit is encrypted by the buyer and then stored by the seller, when a buyer makes a transaction the value of the transaction is sent to the
seller encrypted under the same homomorphic encryption scheme allowing the seller to deduct the value from the balance.

The OT protocols that allow the chooser to actively select a record to be retrieved, and that have linear or sub-linear complexity, can also be referred to as SPIR protocols, as the primitive protects the records of both parties during the information retrieval process. In addition to already discussed uses the OT, and SPIR, primitives can be employed in a variety of systems: electronic watch-lists of suspects [86]; cooperative scientific computation [108, 109]; and on-line auctions [110].

Research presented in [111] implies it is unlikely to achieve an OT without a trapdoor function, which is a public key operation. For this reason most OT protocols are based on the asymmetric encryption employing exponentiation modulo of a prime number. Consequently, most OTs can benefit from a technique for fast exponentiation employed in [107] and defined in Brickell-Gordon-McCurley patent [112]. This technique can greatly improve the performance of most protocols based on modular exponentiation. However, researchers have often chose not to discuss this in detail in the relevant publications discussing the use of PIR and OT primitives, as this is purely a technicality, and it tends to obfuscate the cryptographic solutions being presented.

3.3.5 Private Value Comparison – Locating interesting records

The primitives defined in the previous sections provide techniques to retrieve a record from the sender, without the chooser revealing anything about the record of interest. However, these primitives require that the record is retrieved using an index. Such approach can be justified for protocols designed for information retrieval from online stores, or databases, where directories providing basic information about each and every record are publicly available. However, in the scenario with no publicly available index of the interesting record, such an approach would fail. Thus, there is a need to provide a method for matching (or comparing) the description of the interesting record with the description record held by the sender, so that an index of the interesting record can be identified. This functionality can be provided by the schemes described next.
An efficient technique for value comparison has been described in [113] where it is used in the context of private bidding. It is also suggested that a protocol that allows for comparing two values privately, where the values are the maximum price a bidder is willing to pay for an item and the minimum price the seller is willing to sell for, can allow for on-line haggling, or bargaining, in order to determine a price of an item. A semi-trusted third party is introduced by [113] in order to minimise the communications and computation required by the protocol. This third party is oblivious to the results of the protocol, and it is only trusted not to collude with any of the participants. Thus, an auction house would be a suitable third party for an implementation of the protocol. The protocol compares values bit-by-bit using PIR circuits based on the difficulty of factoring (as per RSA [47]) and higher-residuosity assumption, as discussed in [114].

Privacy-preserving approaches to compare information are essentially different approaches to solve Yao’s millionaires’ problem. While the millionaires’ problem is a good example for an academic discussion, in practice the comparison circuit can be used to facilitate Internet second-price auctions [115], and any other operations where the value comparison must be run on secret inputs. A number of interesting and unconventional approaches to performing data comparison is provided in [116]. The scenario published in [116] requires comparing two secret entries, which in this case is the name(s) of an individual or individuals that made complaints to two managers participating in the protocol, in order to check whether the complaints were made by the same individual. This calls for a special case of value comparison, which is Private Equality Test (PEqT), where in [86] it is referred to as an asymmetric equality test. PEqT is the key primitive in the area of private matching algorithms. This primitive allows two parties to compare their secret inputs for equality without revealing these inputs. There are two cryptographic concepts that the PEqT can be based on: commutative cryptography [50, 61, 116]; or homomorphic cryptography [92, 110]. Each of these techniques has its benefits depending on the problem at hand.

The first published solution to the private matching problem is the commutative cryptography scheme used in the protocol for playing Mental Poker over a distance [61] first drafted in 1979, and further analysed by Shamir in [50]. For two parties,
Alice and Bob, each holding different commutative cryptography keys, the operation of the protocol is summarised in Figure 3-8 and illustrated in Figure 3-9.

**Private Equality Test:**
1. Alice encrypts her input and sends it to Bob,
2. Bob encrypts the ciphertext received from Alice and sends it back,
3. Bob encrypts his secret input and sends it to Alice,
4. Alice encrypts the ciphertext containing Bob’s input,
5. Alice compares the two resulting ciphertexts, if they are equal then her and Bob’s inputs are equal,
6. Alice may inform Bob about the result.

**Figure 3-8 Description of Private Equality Test based on Commutative Cryptography**

This scheme employs a modification of the Pohlig-Hellman (PH) algorithm described further in Chapter 4. Thus, each encryption/decryption operation requires only a single exponentiation. To date, a number of different PEqT schemes have been proposed, but the complexity of the other schemes is usually higher than this of the commutative encryption solution presented above. Boa and Deng [92] described an efficient method for equality testing based on homomorphic encryption. However,
this method requires a series of multiplications, an exponentiation, as well as a round of homomorphic encryption and decryption. The homomorphic encryption used in their scheme is ElGamal, which itself requires two exponentiations modulo a prime during the encryption process, and another for the decryption operation. Consequently, the complexity of their protocol, as well as the protocol described in [110], is higher than the complexity of the PEqT scheme illustrated in Figure 3-9. However, only the slight difference in performance means that the decision of using one or the other method should be based on factors other than efficiency alone.

When two parties engage in equality test protocols, often there are a number of inputs to be compared. Thus, a scenario exists where the chooser, Alice, wants to compare her value with a number, \( n \), of values held by the sender, Bob. In such a scenario, if the homomorphic scheme of Boa and Deng [92] was to be used, then for each record held by Bob, four exponentiations would be required. However, the commutative cryptography-based PEqT shown above can be modified so that the 1-n PEqT protocol could be achieved with \( O(n + 3) \) exponentiations. The modification of the protocol is in Step 3, where Alice computes a value equal to her input encrypted only by Bob. In this way she can now compare the resulting value with Bob’s inputs encrypted by only him. The resulting protocol that allows Alice to compare her secret input with \( n \) inputs held by Bob, involves the steps described in Figure 3-10 and illustrated in Figure 3-11.

As far as the openly available literature goes this protocol is likely to be the most efficient 1-to-n PEqT (or 1-n PEqT) protocol available. Interesting extensions to the concept of 1-n PEqT are private intersection and private intersection size protocols. Thus, this protocol can be extended into the intersection size protocol described [63]. A similar approach to computing secure intersection size is also provided in [117]. Whereas, Freedman, Nissim, and Pinkas presented an efficient secure intersection protocol in [118] that is improved in [119]. Weis argues that these protocols share a fundamental security flaw, as for malicious party it is trivial to convince the other party than an intersection exists [57].
1-to-n Private Equality Test:
1. Alice encrypts her input and sends it to Bob,
2. Bob encrypts the ciphertext received from Alice and sends it back,
3. Alice decrypts the ciphertext containing her input encrypted by her and Bob,
4. Bob encrypts all his secret inputs and sends them to Alice,
5. Alice compares the result from Step 3 with other ciphertexts received from Bob in Step 4, and if equal ciphertext are found, then she knows that Bob has got a common element with her set.
6. Alice may inform Bob about the result.

3.3.6 Combined approaches to selective information retrieval

The Private Equi-Join (PE) protocol can enable two parties, the chooser and the sender, to privately compare their sets of unique values $V_C$ and $V_S$, and allows the chooser to retrieve some extra information $ext(v)$ about records $V_S$, that match
records $V_C$ on a given parameter [63]. The PE protocol involves the steps described in Figure 3-12.

**Private Equi-join protocol:**
1. Both parties apply hash function $h$ to the elements in their sets, so that $X_C = h(V_C)$ and $X_S = h(V_S)$. Chooser picks a secret PH key $E_C$ at random, and sender picks two PH keys $E_S$ and $E'_S$, all from the same group $Z_p^*$.
2. Chooser encrypts entries in the set: $Y_C = E_C(X_C) = E_C(h(V_C))$.
3. Chooser sends to sender set $Y_C$, reordered lexicographically.
4. Sender encrypts each entry $y \in Y_C$, received from the chooser, with both $E_S$ and $E'_S$ and for each returns 3-tuple $(y, E_S(y), E'_S(y))$.
5. For each $h(v) \in X_S$, sender does the following:
   (a) Encrypts $h(v)$ with $E_S$ for use in equality test.
   (b) Encrypts $h(v)$ with $E'_S$ for use as a key to lock the extra information about $v$, $\kappa(v) = E'_S(h(v))$.
   (c) Encrypts the extra information $\text{ext}(v)$:
       \[ c(v) = K(\kappa(v), \text{ext}(v)) \]
       Where $K$ is a symmetric encryption function and $\kappa(v)$ is the key crafted in Stage 5b.
   (d) Forms a pair $\langle E_S(h(v)), c(v) \rangle$. These pairs, containing a private match element and the encrypted extra information about record $v$, are then transferred to chooser.
6. Chooser removes her encryption $E_C$ from all entries in the 3-tuples received in Step 4 obtaining tuples $\alpha$, $\beta$, and $\gamma$ such that $\langle \alpha, \beta, \gamma \rangle = \langle h(v), E_S(h(v)), E'_S(h(v)) \rangle$. Thus, $\alpha$ is the hashed value $v \in V_C$, $\beta$ is the hashed value $v$ encrypted using $E_S$ and $\gamma$ is the hashed value $v$ encrypted using $E'_S$.
7. Chooser sets aside all pairs received in Step 5, whose
first entry is equal to one of the \( \beta \) tuples obtained in Step 6. Then using the \( y \) tuples as symmetric keys it decrypts the extra information contained in the second entry in the pair \( \{E_2(h(v)), c(v)\} \).

**Figure 3-12 Operation of the Private Equi-join protocol**

Researchers have shown that using a TC device such as PCI-attached IBM’s 4758 SCOP, it is possible to perform efficient hardware-based PIR that allows for the selection of a record based on given match criteria [78]. In such a scenario SCOP can easily match any record based on selection criteria, however, the problem is still in retrieving the record in a way that the host computer cannot identify the record that is sent back to the chooser. In an ideal scenario the SCOP would collect a number of records stored on, or accessed through, the host machine so that is impossible to identify which record is being send to the chooser. However, the difficulty lies in the fact that SCOPs often do not have enough memory to store and process many records, as they have a limited amount of RAM. The solutions presented in [78] involve SCOP performing the steps detailed in Figure 3-13.

**Hardware PIR:**

1. Retrieve records one-by-one.
2. Compare each record to the match criteria.
3. Encrypt each record and store in the host’s memory system, keeping a note of the memory location belonging to the record matching the selection criteria.
4. Once all the records have been retrieved by SCOP and stored encrypted in the host’s machine, shuffle and re-encrypt all records. In this way the host machine can no longer link records retrieved in Step 1 to their encrypted form.
5. Pick the record matching the selection criteria and send it securely to the requestor.
Introduction of the hardware into the PIR has not greatly lowered the complexity of this primitive. SCOP needs to encrypt each record at least twice (Step 3 and Step 4), and the operations related with loading and unloading the data from the SCOP also add a delay into the protocol. By introduction of the \textit{square-root} algorithm \(i\) different PIR requests can be allowed to run following a single shuffle of the records. This improves greatly the performance of the protocol, but, in order for the \textit{sender} not to realise whether a given record has already been requested or not, the SCOP needs to pick from the host’s storage every record that already has been picked in a given shuffle round, as well as one new record. This should be either the record selected by the \textit{chooser}, or a randomly selected one if the \textit{chooser} requested a record that has already been picked in this round. A new shuffle round is run every \(i = \sqrt{n}\) records [78]. It is worth noting that this is a PIR and not SPIR protocol, since there is no way for the \textit{sender} to judge how many records have been retrieved.

### 3.3.7 Security Considerations

Security measures should always be considered in relation to realistic threats to a given system. Thus, Goldwasser discusses four adversaries specific to MPC protocols [100], these are:

- **Passive.** One or more participating party aiming to obtain the secret input of the other participants.

- **Byzantine.** Otherwise referred to as malicious adversary. A party that does not follow the protocol, and provides other parties with specially crafted inputs in order to obtain secrets or compromise other participating parties in another way.

- **Mobile.** A coalition of the passive and byzantine adversaries formed by different set of parties at each round of the MPC protocol.

- **Coercing.** This can force users to provide specific inputs, for example vote in a specific way during electronic elections.

The security of a cryptosystem often depends on correct design and implementation, just as much as on the strength of underlying cryptographic protocol [120]. Most attacks on common systems are possible only due to erroneous design, implementation or maintenance, and not weaknesses of the underlying cryptographic
algorithms [121]. Consequently, just because a given cryptographic protocol used strong underlying algorithm, it does not mean that the protocol is secure as shown by the example of a watermarking protocol that fails to sufficiently link the watermarks to the digital goods being signed described in [51]. Because of the nature of cryptanalysis, where any cryptosystem or ciphertext may be attacked in an arbitrary way by a previously unknown attacker, to the cryptography is being compared to *programming Satan’s computer* [122]. The key lesson to learn from this approach to cryptography is that it is likely that errors in implementations of certain protocols will occur, and there needs to be a mechanism that will allow for these errors to be fixed (this is also highlighted in [123]).

When RSA was first published its authors encouraged the readers to attempt breaking their algorithm, as they wanted to make sure that they have not overlooked any potential flows in the system. RSA was the first cryptosystem *trap-door one-way permutation*, and thus the exact strength of the algorithm was not known [47]. Nowadays, over 30 years later, RSA is still considered as secure if certain conditions are met. RSA has been an inspiration for a number of privacy-preserving solutions. Over the years its security has been addressed by a number of academic and industry studies. These have been summarised in [124], where the *RSA Problem* has been formally defined as the problem of obtaining the plaintext message from the ciphertext and the public key used to produce this ciphertext. It is shown that the RSA Problem is no harder than integer factoring, however, taking into consideration that RSA modulus $n$ is sufficiently high and randomly generated, then RSA Problem is hard to solve. However, the randomness of the plaintext over the range $[0, I-1]$ is also crucial. Some studies of the RSA suggest that using strong primes in the algorithms is necessary in order to safeguard the systems from factoring and *cycling* attacks. However, Rivest and Silverman prove that using strong primes in RSA yields limited benefits to the strength of the protocol, as long as the primes used are reasonably large [125]. This is unlike another protocol commonly employed in MPC, the DH, and consequently ElGamal needs to be based on strong primes [121]. In [126], Sakurai and Shizuya discuss the security of various protocols (including DH, 3Pass, and ElGamal) based on the Discrete Logarithm Problem (DLP). Their research suggests that ElGamal and DH can both be reduced to 3Pass in a polynomial time, and all three protocols should be considered as equally strong. No
efficient attacks against DLP are currently known, however, this does not necessarily
mean that the schemes based on DPL cannot be broken without breaking the DLP
[126].

A number of primitives described in this chapter assume an existence of a secure
commutative encryption scheme. However, traditional means of testing the security
of encryption schemes are not capable of evaluating schemes where ciphertexts
commutate [57, 127]. In [127], a new model for assessing such protocols is
presented, and the security of commutative protocols based on RSA is shown to be in
NP. Additionally, [57] presents a technique that can transform any semantically
secure homomorphic encryption scheme into semantically secure commutative
scheme, thus, allowing already existing crypto libraries to perform commutative
operations. Finally, in Section 2.5.4, Massey-Omura cryptosystem was suggested as
one of possible algorithms that could be employed in systems relaying on
commutative cryptography. This cryptosystem performs operations in GF(2ⁿ) field in
order to allow hardware-accelerated implementation of the Pohlig-Hellman
cryptosystem. However, [128] shows discrete logarithms are easier to compute in
this field than in GF(p), and therefore using this field for cryptographic operations
should be carefully considered.

It is possible to cheat in some protocols and provide the other party with crafted input
data that has not been created using an encryption key on the originator, but prepared
in order to reveal the secret of the second participant. Such a scenario can be
mitigated by the use of ZKP on the inputs from another party, just to prove that the
inputs have been generated according to the protocol [85]. In academic discussions
and few specific real-life scenarios, it is possible to ignore the threat from the
possible exploit by assuming that the participants are honest, but curious. Often PET
protocols are presented in honest-but-curious form in order to simplify the analysis
of the protocols [85, 86, 103, 115]. This assumes that the participating parties follow
the protocol (honestly) but will try to compute and imply any information they can
with any data obtained during the process (curiously). These protocols can then be
transformed into malicious mode with use of ZKPs.

Another security problem in complex systems is the fact that a number of privacy-
preserving primitives may need to be used in order to perform a given task. Such
composite system would reveal more information than required, since apart of the final output the intermediate results would be revealed. As [117] suggests, it is possible to define the intermediate results as a part of the output, in order to evaluate a system under the rules of MPC. This is not an ideal solution, thus, such composite systems should be avoided if possible. However, this solution ensures controlled disclosure and, in most cases, this is sufficient.

3.4 Conclusion

In theory the DPA provides individuals using UK-based services with a full control of their personal data. An organisation wishing to collect data about an individual must obtain consent (this can be implicit) from this individual. This consent can be withdrawn by the individual/data subject at any time, by an opt-out procedure that all organisations storing personal data must provide. However, even if a data subject perceives a given system as intrusive the convenience and economical factors can force this data subject to keep using the system. In such scenarios the individuals often decide to use anonymising technologies on the Internet, and tend to use cash in face-to-face transactions. The anonymising technologies, include TOR, are created by a large number of users creating a virtual network (over the Internet) that can hide the identity of an Internet user among 100,000 other TOR users. It uses onion routing based on sequential re-encryption of network packets in order to stop the ISPs and other TOR users from tracing the network packets back to the user that generated them. While TOR is implemented, and is commonly known to the Internet users interested in maintaining their privacy, there are other system proposed that can allow for anonymised purchases, and on-line auctions. These usually rely on concepts from the area of MPC.

MPC protocols can also be used to facilitate privacy-preserving investigations. Literature shows that it is possible to build systems that allow investigators to trace data subjects marked as suspects, without revealing the identity of the suspect or affecting the privacy of other data subjects in a given system. Also, it is possible to create pseudonym-based auditing systems that only reveal the identity of an individual if the actions performed by this individual have reached threshold of malicious activity.
The solutions employing MPC are often impractical. In early MPC protocols the computational complexity was exponential to the number of bits used to store the private records and to the size of the data records. Fortunately, it is possible to manipulate computational and communicational complexity of different schemes by using different cryptographic techniques and introducing semi-trusted third parties that could proxy the requests. In similar fashion to PKI, MPC often employs symmetric encryption to lock data that is transferred between the participating parties, while computationally-expensive trapdoor functions, such as public-key cryptography, are used only to conditionally exchange the symmetric encryption keys. It is worth noting that it is seldom for the MPC-related research to provide empirical evaluation of protocols. Most research into MPC has focused on perfecting previously developed schemes, with little attention paid to their practical use \([103-105, 107]\). A comparison of the different schemes is usually done on the basis of computational and communicational complexity, which, some researchers assert should not be directly compared. In general the efficiency of encryption schemes based on modular exponentiation (used by most trapdoor functions) is approx. 0.1% of the symmetric encryption protocols. Thus, a protocol that takes \(O(1000 \times n)\) symmetric operations, would take a similar amount of time to a protocol with \(O(n)\) trapdoor function operations, while the computational complexity expressed in terms of the number of operations would suggest otherwise.

While there is a number of PIR and OT primitives that allow private retrieval of records, most require the interesting record to be identified by its index in a given dataset. This approach is optimal in scenarios with an index or a catalogue of the database being available publicly. Such scenarios include purchasing goods or services from an on-line retailer or a service provider, which is the main motivation for a number of PETs. However, in order to retrieve records matching certain selection criteria, it is necessary to run equality tests on the data. It is possible to combine a PEqT primitive with PIR or OT in order to achieve such functionality. But it is also suggested that complex MPC protocols made up of a number of privacy-preserving primitives can release more information than required by a given scenario and the complexity of such protocols is usually less optimal than custom-made protocols. However, the PE protocol based on commutative cryptographic algorithm is a purpose built system for retrieving records that match given selection criteria. Its
authors suggest that it should be suitable for use in sharing data between hospitals and other organisations with large databases. It is also possible to achieve similar system if TC hardware device SCOP is deployed to the database server (or a host attached to the database), but it is much harder to provide guarantees of privacy of other data records stored by the dataholder if SCOP is deployed.
Chapter 4

Improving the Acquisition Process

4.1 Introduction

This chapter documents the initial evaluation of the PET protocols in an investigative context. For this purpose a set of requirements for data acquisition process is drafted and refined. These are based on the available literature such as regulations, guidelines and procedures. Often these requirements are inferred rather than obtained from these sources, and thus expert opinion was obtained as to their validity (see Section 4.7.3 Feedback from practitioners).

Gathered requirements are used to select candidate PET primitives. Two different approaches to building a suitable solution are proposed based on related research. These include a solution built from 1-n PEqT and 1-n OT, as well as a solution based solely on the PE primitive. The approaches are empirically evaluated based on tables of the computational complexity produced for each solution and experimentally
established timings for applicable cryptographic operations. Advantages and disadvantages of both solutions are identified based on the requirements.

The outcome of this chapter is a set of requirements for data acquisition process and the suggestion of a protocol capable to satisfy most of these requirements. This contributes towards the design of the data acquisition platform proposed in this thesis and presented in Chapter 5.

4.2 Methodology

Chapter 3 concluded that it is possible to protect the interests of two parties wanting to compute a function without revealing the secret inputs or to conditionally exchange some data between the parties, with the use of PETs. Arguably, it should be feasible to construct a process that employs PETs to retrieve investigative data in a privacy-preserving manner. With the range of PIR, OT and SPIR primitives available, it is likely that a single protocol can perform the required operation, and, if not, then a combination of existing primitives should be able to achieve this. However, before such primitives can be identified, a set of requirements for the data acquisition system needs to be drafted. This can be done based on the literature discussing the data acquisition process, UK legislation and digital forensics research. The gathered requirements can then be used to define evaluation metrics for the platform and to analyse the available PET primitives for the suitability of use in an investigative scenario. Thus, the identified primitives can be evaluated more thoroughly, and compared against each other. Summarising the methodology of the work presented in this chapter is as follows:

- Define the requirements for a data acquisition platform.
- Identify the evaluation criteria.
- Select the types of protocols that can improve privacy in data acquisition process
- Evaluate known protocols in order to select the most suitable.

It should be noted that at this stage that only existing primitives are taken into consideration.
4.3 Initial Requirements

The requirements for a data acquisition platform that can be derived from Chapter 2 and Chapter 3 can form a guideline for design, implementation and evaluation of a data acquisition platform. For example there is a suggestion that the Internet is now used in organised crime [2] which can mean that some inquiries will require the retrieving of data about a group of suspects, rather than a single individual. Consequently, the protocol chosen for the data acquisition process will need to allow for the retrieval of a number of interesting records at the time, or, if this is not the case, multiple sequential runs of the protocol should bear low computational and communicational overhead. Also, investigators need to provide justification for the acquisition requests under the DPA and the dataholder can refuse to provide any data without a valid warrant from the court of law [10]. Whereas, data acquisition notices served under RIPA do not need any form of justification to the dataholder and the dataholder will face penalty if the relevant data is not provided to requesting public authority within two weeks. Still, the dataholder may choose to accept the penalty and refuse to provide any data without a subpoena [8, 17]. Consequently, the platform must leave the dataholder in control of the data, since the data retrieval can only be performed with the dataholder’s consent. Taking into consideration that a dataholder has two weeks to provide the data under RIPA the computational complexity of the protocol can be reasonably large [17]. However, shortening the time required by the data acquisition process is one of the main reasons provided in [2] as a justification for the proposed modernisation of the data acquisition process.

Any evidence collected may need to be presented in front of court of law, which will require that the electronic evidence must be provided as a true image of the data gathered. So data records should be retrieved from the dataholder on a record-by-record basis, so that if only one of many records is required for the investigation, other records can be discarded. Otherwise the public authorities can end-up storing large amount of unnecessary data, and this can prove costly, taking into consideration the level of security and auditing involved in storing digital evidence [20]. All processes applied to computer-based electronic evidence should be preserved in an audit log so that an independent third party could examine these processes and achieve the same result [20].
Under RIPA the public authorities must make a contribution towards the costs incurred by a CSP during the fulfilling of the data acquisition notice [17]. Thus, the cost of the solution should be low. If the costs were not covered by the authorities, the dataholders would transfer the costs of handling the enquiries to the end-users and such a solution would typically be unacceptable by society. Additionally, to put this task in the context, BT has 15 million broadband users [129], and thus, the system developed in this thesis must handle datasets of similar size.

To summarise the above discussion a platform for digital acquisition needs to meet the following requirements:

**Req. 1** Allow for the gathering of multiple suspect records per enquiry, or have low overhead per each additional query run on the database.

**Req. 2** Keep the data controller in charge of the data. A data record cannot be transferred or made available, to the public authorities, without the data controller’s verification of the request.

**Req. 3** Allow for efficient and timely data retrieval. The current maximum time for returning data under RIPA is two weeks, however, it is expected that in urgent enquiries, investigators would have access to data in a reasonably short time.

**Req. 4** Be cost-effective, as the platform will need to be deployed by a variety of organisations.

**Req. 5** Retain an audit trail of the processing performed on the potential evidence.

**Req. 6** Gain acceptance from the general public.

**Req. 7** Handle large datasets (such as datasets with more than 15 million records).

### 4.4 Overall design

According to [8] two fundamental parts of data acquisition process are: serving the notice to the dataholder; and the subsequent retrieval of data. The retrieval is often achieved by the dataholder sending back the data to the requestor. The request and the response both need to be performed under the guidelines of the DPA. Thus, some form of secure channel needs to be established between the parties, or the messages need to be encrypted while in transit, with a technique that is either FIPS140 compliant or at least FIPS198 compliant. Currently, such a data acquisition notice would include the specification of the requested records that the dataholder would then use to build a database query. Almost all rational databases support Structured
Query Language (SQL) queries [130] and most likely any data acquisition notice would be translated into such a query. Thus, the notice and the SQL query should contain the following parameters:

1) Identification of the type of the information that is required. These could be number parameters that contain answers to investigator’s questions. (Represented in SQL representation introduced below as $H$ different Return Parameters, $rp_1 - rp_H$.) For example in an enquiry for the recent use of a given credit card, the return parameters would consist the location where the card was last used, together with the transaction amount and the date of the transactions.

2) Specification of any circumstantial request constrains. (Defined in this thesis as $L$ different Input Parameters, $ip_1 - ip_L$, with values $ip\_val_1 - ip\_val_L$.) Using the above scenario this could include the time constrains specifying the time window for interesting transactions, such as “all transaction between 12/08/2010 and 18/09/2010”.

3) Specification of the relevant data subject, the individual whose data is being retrieved, by providing the ID of the interesting record (such as the mobile phone number of the suspect). (This parameter is later referred to as the Record of the Interest $ri$, with value of $ri\_val$). In the above scenario this would be the credit card number.

Then, if we refer to the dataset as the source, the request for investigative data could be mapped into the following SQL query:

```
SELECT rp_1, rp_2, ..., rp_H
FROM source
WHERE ri = ri_val AND ip_1 = ip\_val_1 AND ... AND ip_L = ip\_val_L
```

Figure 4-1 Typical request for investigative data mapped into SQL

In most cases the names of the return parameters, as well as the names of the input parameters, and values of these input parameters, can be openly communicated. But the value of the interesting record ($ri\_val$) is used to uniquely identify the suspect and therefore in order to provide privacy to the potential suspects, it must be hidden. This can be achieved by running a database query for the return parameters of all the records that satisfy the conditions defined by the input parameters, and then
collecting the interesting record from the dataholder using the some privacy-preserving protocol based on the OT primitive. Consequently, the query that is actually run on the dataholder’s database can be rewritten as:

```
SELECT ri1, rp1, ri2, ..., rpH
FROM source
WHERE ip1=ip_val1 AND ip2=ip_val2 AND ... AND ipL=ip_valL
```

Figure 4-2 Request enabling privacy-preserving queries mapped into SQL

Chapter 3 concluded that in order to retrieve data in a privacy-preserving manner there needs to be a publically available index of data records or 1-n PEqT protocol needs to be run in order to obtain such index of (or other form of pointer to) the records of interest. Only then the OT or SPIR primitive can be used to retrieve data from the dataholder. Consequently, there are three distinct operations required in the process of data acquisition. For simplicity in this, and further consideration of the scenarios, a relevant public authority requesting the data is referred to as the chooser, while a dataholder is called the sender. Thus, the following are the key operations needed to acquire investigative data in a privacy-preserving manner:

1) **Querying.** The chooser specifies the type of information that is required for the investigation. This can be achieved using SQL, since it provides standardised format for database querying.

2) **Searching.** Allows the chooser to find an index of, or a pointer to, the interesting record in the sender’s database, by the means of private-matching techniques, such as PEqT.

3) **Retrieval.** Finally, the interesting record is retrieved from the sender using the OT or SPIR primitive.

The above list excludes some crucial elements of the data acquisition process that are derived from RIPA, DPA and the guidelines on data acquisition [8, 10, 17]. Those excluded processes are the steps required to obtain authorisation and the definition of the roles in the data acquisition process (Applicant, Designated Person, SPoC, or Senior Responsible Officer). They have been discussed in Chapter 3, and it has been established that these processes are fit-for-purpose and protect the integrity of the investigation and privacy of the involved parties as much as possible, without the
involvement of PET technologies. On the other hand, PET technologies appear to be capable of fitting into this process smoothly, by replacing the current technologies used during for exchanging notices and the data between the SPoCs of the relevant parties.

There are two possible solutions to address searching and retrieval operations. These can be achieved using a combination of PEqT and SPIR, but also a combined approach, such as PE, can be used. There are advantages to both of these approaches. Using a combination of primitives it may be possible to keep more detailed audit logs and provide verification for requests (Req. 5 and Req. 2), as the searching phase is independent and this would potentially allow for running independent checks on the records being requested by the choosers. Also, such a solution could prove to be the least costly (Req. 4) as some of the primitives are built using standard cryptographic protocols that can be found in existing cryptographic libraries, which would cut the development, compliance testing and maintenance costs, and also make the solution more transparent (contributing towards Req. 6). On the other hand a combination of different privacy preserving primitives can reveal some extra information that is needed to link the two primitives [117], which is not the case if a problem-specific solution is derived straight from cryptographic algorithms. The downfall can be reduced, however, the mitigation can increase the complexity of the protocol, and hence its cost and time required for queries would also increase. Therefore, since most problem-specific approaches are usually more efficient than a combination of two primitives, it is likely that a protocol such as PE would better fulfil Req. 3.

It is often impossible to compare protocols based on theoretical evaluation criteria such as communicational and computational complexity. These parameters can only be used to compare the efficiency of protocols built in a similar way, thus an improved version of a protocol and an original version of this protocol can be directly compared using these parameters. But protocols built on different concepts can not be directly compared in this way [105]. Consequently, in order to find a suitable protocol for data acquisition purposes, it was necessary to find a good combination of PET primitives that can perform actions similar to those required by the requirements, and also a combined problem-specific primitive that matches most
closely the requirements, and then compare these two approaches. Section 4.5 describes the design of the Searching and Retrieval functionality for the data acquisition framework with use of separate primitives for these functions, while Section 4.6 provides details of a design based on a problem-specific primitive. In Section 4.7 these designs are evaluated side-by-side against the requirements.

4.5 Approach 1: Combination of PET primitives

Both Searching and Retrieval phases of the process can be performed by a number of different primitives. These are analysed and suitable candidates for implementation are selected. These candidates are then put together to provide the required functionality as described by the Design and Implementation section. The selected solution has been published in [9].

The Searching phase needs to establish a pointer to the interesting records in the database, or, more precisely, to the source table resulting from the Querying phase. In practice this can be achieved by privately comparing the identity of the interesting record to the records in the source for equality. From the protocols discussed in Section 3.3.5 the one that truly stands out as the most likely to be efficient 1-n PEqT protocol is derived from 1-2 PEqT presented by Shamir et. al. in [61] and illustrated in Figure 3-11.

Some homomorphic encryption based protocols, such as the one presented in [92], can compete with the efficiency of the commutative solution in 1-2 PEqT operations. However, such protocols normally need to be completely rerun for each record being compared, and, thus in 1-n PEqT, the complexity would simply be increased by n times. In the case of the protocol derived from [61] this is not the case and while the computational complexity of the 1-2 PEqT based on commutative cryptography is \( O(4) \), the derived 1-n PEqT is characterised by computational complexity of \( O(n + 3) \). This would satisfy Req. 1 and Req. 3 that deal with efficiency and rapidness of the enquiry. Since, the protocol can be based on a specific case of RSA or Pohlig-Hellman algorithms, the development time necessary would be minimal and the solution should be easier to explain to decision makers that are aware of current encryption standards (which is key to gain acceptance for the protocol). This is the case where the protocols are well known, and thus well researched. Thus, Req. 4 and
Req. 6 would most likely be satisfied as well. The *sender* would not have to disclose any information apart from the confirmation whether the interesting record exists in the *source*, and the location (or index) of this record.

The selected 1-n PEqT is based on commutative cryptography, and thus a suitable commutative cryptography protocol has to be selected first. Section 2.5.4 discussed different commutative algorithms. From those protocols the one based on ElGamal encryption cannot be employed in the 1-n PEqT test, as the plaintexts encrypted under two different keys used in arbitrary order are not equal under ElGamal. Also, the literature suggests that Pohlig-Hellman is a better choice than Massey-Omura cryptosystem, since the discrete logarithm problem is harder in the GF\((p)\) field than it is in GF\((2^n)\) [128] (as discussed in Section 3.3.7). Thus, the protocol selected for the implementation needs to be either the Shamir’s commutative protocol (based on Pohlig-Hellman algorithm) or a modification of the RSA scheme, sometimes referred to as Shamir-Rivest-Adleman (SRA) protocol [127]. Since, the OT selected for use in the Retrieval can be based on RSA, SRA is chosen. This means that the solution could be fully implemented using common cryptographic suites such as Legion of the Bouncy Castle cryptography library [131], with small changes to the way that RSA keys are exchanged, as in SRA both the encryption and decryption keys need to be kept private, and only the modulus, and the primes used to generate it, are shared between the parties.

Most OT and SPIR protocols can be used to perform the Retrieval phase and the difference is mainly in performance. One of the protocols that stands out for its use of common encryption protocols in the design is the OT discussed by Schneier [44] and presented in Figure 3-7. Schneier’s example illustrated 1-2 OT protocol; however, it can be extended to perform 1-n OT functionality. It can be built using a combination of virtually any type of public-key and private-key encryption algorithm. Thus, it would be possible to obtain FIPS 140 accreditation for the Retrieval part of the data acquisition process. Also, this OT protocol is relatively easy to comprehend by the professional audience, including the decision-makers, with basic understanding of PKI. Therefore, such a solution could be presented to the relevant decision makers regulating the investigative data acquisition field.
In fact, the operation of most OT protocols requires that the sender provide the chooser with an encrypted copy of all records in the table resulting from the Querying phase. Then, the control over the interesting record (its decryption key) is retrieved using an OT protocol. The operation of the chosen 1-n OT protocol is described in Figure 4-3. The choice of the asymmetric public key encryption is limited by the fact that the encryption protocol cannot issue errors if wrong key is used to decrypt a ciphertext. Thus, ElGamal cannot be used, but RSA is a good choice, especially that it ties-in with the SRA used in the Searching phase. On the other hand, this OT protocol can be based on virtually any symmetric encryption primitive, despite the fact that similar restrictions (key verification) are put upon the symmetric operations in the process. This is due to the fact that key verification is seldom implemented in symmetric encryption algorithms and if it exists it is usually part of the protocol implementation, and not the actual maths used in the algorithm. Thus, AES has been chosen as per current industry standards and FIPS 140 specification.

<table>
<thead>
<tr>
<th>Approach 1: Retrieval Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The sender generates n sets of public/private keys pairs, and sends all public keys to the chooser, preserving the order in which they have been sent.</td>
</tr>
<tr>
<td>2. The chooser generates a key with a private encryption algorithm, such as AES, later called AES key. It then uses the $i^{th}$ public key received from the sender in Step 1 to encrypt the AES key and send it to the sender.</td>
</tr>
<tr>
<td>3. The sender does not know which public key has been used to encode the AES key, or which record has been selected, thus protecting the privacy of the suspect. The sender can then decode the cipher-text received in Step 2 using all private keys generated in Step 1, whilst preserving the order in which they have been decrypted. In this way $n$ potential AES keys are created. Only the $i^{th}$ one is the proper AES key; the other outputs are random sets of bits, which cannot be distinguished from ordinary AES keys.</td>
</tr>
</tbody>
</table>
4. The sender encrypts all records using appropriate keys decrypted in Step 3. Thus, the first record in selected records is encrypted with an AES key decrypted using the first private key generated in Step 1. Consequently the $i^{th}$ record, which includes data about the suspect, is encrypted using the AES key generated by the chooser in Step 2, sent to the sender encrypted by the $i^{th}$ public key, and then decrypted using $i^{th}$ private key. In this way the $i^{th}$ record will be encrypted using the proper AES key.

5. The chooser gets $n$ encrypted records, but using the AES key it is able to decrypt only the $i^{th}$ record. Other records are unreadable to the chooser provided that the false keys generated in Step 3, and used to encrypt these records in Step 4, are not broken.

Figure 4-3 Retrieval Phase

4.6 Approach 2: Combined PET primitives

Protocols created for the purpose of searching datasets and retrieving objects of interest in private-manner do exist. These are examined in this section and a suitable protocol is selected and put forward for comparison with a solution made from a combination of PEqT and OT primitives (presented in Section 4.5).

Some pseudonym-based systems, such as those proposed by Biskup and Flegel in [87, 88], provide adequate functionality and could, in theory, fulfil the requirements of the data acquisition process. In fact, a solution based on pseudonyms would most likely gain the acceptance of Society, since it is an easy to comprehend approach that can provide information theoretic security for the parties involved. However, it would not meet Req. 7, as it does not scale well, and would be impractical for a system with large amounts of records. The protocol for private on-line transactions presented in [85] provides some functionality of what is required by the data acquisition process. It allows for the retrieval of digital goods based on publicly available index; however, it also provides a private comparison functionality that
ensures that the buyer has enough funds to purchase the goods. Therefore, it would be possible to modify this protocol in order to create an adequate data acquisition protocol. On the other hand, the PE protocol provides all the basic functionality required, and it is designed to handle multiple records in the request (which would help to satisfy Req. 1). Consequently, the PE protocol has been chosen as the suitable combined approach.

The operation of the PE protocol is described in Figure 3-12. It uses three different commutative keys to facilitate Searching and Retrieval. Identifiers of the records of interest are hashed by the chooser, while the identifiers of the records resulting from the Querying phase are hashed by the sender, and then compared using commutative 1-n PEqT primitive. The records themselves are encrypted under symmetric encryption with keys crafted from the hashed identifiers encrypted using commutative keys. The chooser then retrieves all the records in the dataset, uses hashes of the identifiers commutatively encrypted by the sender to locate the records of interest, and decrypts these records using keys obtained in a fashion similar to the 3Pass primitive. Thus, the protocol requires two different forms of encryption: commutative and symmetric encryption. As previously defined, AES is a good choice for the symmetric cryptographic operations. However, in this approach the design is based on commutative properties of the PH protocol and not RSA, as public-key functionality is not required and PH cryptosystem does not provide the additional avenues for attack, namely the large-number factoring problem, that RSA is based on.

4.7 Evaluation

4.7.1 Experiment Design and Implementation

For the purpose of performing an initial evaluation on the different ways to implement the PET solution for the data acquisition process, proof of concept testing needed to be performed in order to establish which approach fulfils the requirements. The requirements that could be considered without experimentation have been discussed already in Section 4.5 and Section 4.6. The key purpose of this empirical evaluation is to allow the comparison of protocol performance. Thus, the two key performance factors that need to be evaluated are: the time required for
computations; and the amount of communication taking place. Typically, such protocols are evaluated using the notions of computational and communicational complexity. However, as discussed earlier, the notion of computational complexity cannot be used where the time for a single operation is different between the protocols. This is clearly the case in the protocols shortlisted, as the first makes use of public-key cryptography, and the other does not. Consequently, some measurements are required to establish the average times for the operations in the process as well as the total time for the protocol run. These are evaluated on a single machine acting as both the chooser and the sender, with a single database to store the records. This choice is natural for empirical evaluation of OT and SPIR protocols since balancers can be used to distribute the load between the participating parties [102, 104] and semi-trusted parties can be used to take-up some of the computational burden [113]. Consequently, performance does not have to be measured on a per-party basis, and only the total time required to reach the result is needed to evaluate the performance for given set of input parameters (as shown in [90]). During the performance testing, the processes were organised into a series as not to affect each other. This means that some optimisation can be added to speed-up the operation of the protocols, but the measurements illustrate the worst-case scenario, and thus allows for direct comparison of the two approaches.

The main variables in the experiment were the size of the dataset being queried \( n \), and the number of interesting records \( m \). In the experiments the output from the Querying on the sender's dataset is simulated by a data table containing 128-bit MS SQL Globally Unique Identifiers (GUIDs) acting as record identifiers \( \text{ri} \) and randomly assigned text of 1kB in size that acted as the data content – the information about the record (as illustrated in Figure 4-4). The test script has randomly selected \( m \) different GUIDs prior to the simulation, in order to act as the identifiers of the interesting records requested by the chooser.

Between the two approaches there are four encryption techniques and one hashing algorithm required to build the experiments. Since, it is not advisable to create an own implementation of encryption protocols [70], where tried and tested crypto suites exist, Bouncy Castle API is employed as the basis for the test implementations. The proof-of-concept protocols themselves are implemented using the C# .NET
programming language in order to speed up the development. Since, both approaches are developed using C#.NET the fact that there are other languages, such as C++, that can produce applications performing faster is irrelevant to this demonstration.

<table>
<thead>
<tr>
<th>GUID</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>0E138AC0-BD34-40DC-A1FB-0000238D746B</td>
<td>Cras nec tellus elit. In hac habitasse platea dictumst. Proin lectus elit, molestie sit amet iaculis quis, consectetur in metus ...</td>
</tr>
<tr>
<td>19E5B1CF-F6FC-41DB-9779-0000562A56A7</td>
<td>Phasellus pulvinar consectetur metus, vel auctor magna malesuada auctor. Suspendisse potenti. Donec eu leo non diam ultricies eleifend. Cras sed lorem elementum erat auctor egestas in at nulla ...</td>
</tr>
<tr>
<td>4629E748-7A74-42D4-9D5C-00006633D3EC</td>
<td>Donec et neque dui, at volutpat urna. Praesent ipsum sapien, laoreet quis tincidunt at, semper at ante ...</td>
</tr>
<tr>
<td>7FA90D8F-40E7-44F2-BBF4-00008DAADC75</td>
<td>Aliquam interdum lectus sagittis mauris sodales sodales. In id aliquet elit ...</td>
</tr>
</tbody>
</table>

Figure 4-4 Test Dataset

**SRA Implementation**

An advantage of SRA over other commutative encryption protocols is that they can be implemented using common cryptographic suites, with only small changes necessary. In SRA and RSA, the encryption (Eqn. 4-1) and the decryption (Eqn. 4-2) operations are identical. These are performed modulo $n$, which is a product of two large primes $p$ and $q$. From these primes $\varphi(n)$ is produced (Eqn. 4-3), which is used to generate the encryption keys. The encryption exponent $e$ is generated randomly from the range shown in Eqn. 4-4, and so that $e$ is co-prime with $\varphi(n)$. Then the decryption $d$ is calculated as the multiplicative inverse of $e$ modulo $\varphi(n)$ (Eqn. 4-5).

All this is identical for SRA and RSA with the exception that in RSA the parties share their public keys that consist of the encryption exponent $e$ and modulus $n$, but keep $\varphi(n)$ and the decryption exponent $d$ secret, whilst in SRA both primes $p$ and $q$ are shared, or even public, and the both exponents need to be kept private.

\[ C = M^e \mod n \quad \text{Eqn. 4-1} \]
\[ M = C^d \mod n \quad \text{Eqn. 4-2} \]
\[ \varphi(n) = (p - 1)(q - 1) \quad \text{Eqn. 4-3} \]
\[ 1 < e < \varphi(n) \quad \text{Eqn. 4-4} \]
In order for the keys to commutate they need to be generated using the same primes, and therefore the crypto suite needs to be modified to accept the primes as inputs to the key generation process, which in most crypto suites is performed by an atomic procedure. In the case of the Bouncy Castle library, the RSAKeyPairGenerator.cs class from the Org.BouncyCastle.Crypto.Generators had to be modified to achieve this.

**C**ommutative **P**H

RSA is based on the PH protocol, and thus there is some deal of similarity between them. However, PH algorithm, does not support public-key operations, as the decryption key can be easily calculated from the encryption key. Nor is it a symmetric algorithm, as two different keys are used for encryption and decryption. Therefore, PH can be considered as an asymmetric private-key encryption algorithm, and this can explain why PH cannot be found in any openly available cryptographic suite. However, thanks to its common elements with RSA, only small modifications are required to the cryptographic suites. Eqn. 4-6 and Eqn. 4-7 show PH encryption and decryption functions respectively.

\[
d e \equiv 1 \mod \phi(n) \Leftrightarrow d = e^{-1} \mod \phi(n)
\]

\textbf{Eqn. 4-5}

\[
C = M^e \mod p
\]

\textbf{Eqn. 4-6}

\[
M = C^d \mod p
\]

\textbf{Eqn. 4-7}

Both operations are performed modulo of a large prime \(p\), and different keys, exponents, are used for encryption (exponent \(e\)) and decryption (exponent \(d\)) in this algorithm. Thus, the main difference between RSA and PH is that RSA uses modulus made of a product of two primes, while PH uses only a single prime \(p\) for the modulus. Consequently, the RSA engine can be used to perform the operations, if the prime \(p\) is provided as an input instead of \(n\). In addition, the encryption exponent \(e\) is randomly chosen in a way analogous to the RSA exponent, with the difference being that the upper limit of the range for \(e\) is different, and that \(e\) needs to be co-prime with \((p - 1)\):
1 < e < (p – 1) \hspace{1cm} \text{Eqn. 4-8}

Then, exponent \(d\) is calculated as:

\[
de \equiv 1 \mod(p-1) \Leftrightarrow d = e^{-1} \mod(p-1)
\] \hspace{1cm} \text{Eqn. 4-9}

Unlike RSA, but just like in SRA, it is easy to calculate the decryption key from the encryption key, thus, \(e\) and \(d\) must remain secret. Again, the modifications necessary to implement PH protocol with use of a crypto suite are limited to the generation of the keys that in the case of the Bouncy Castle library need RSAKeyPairGenerator.cs to be modified.

### 4.7.2 Empirical Evaluation

The Bouncy Castle crypto library has been used to produce implementation of RSA, SRA, PH and AES encryption schemes, as well as the SHA-256 hashing protocol. These implementations are used to gather performance data relating to the generation of cryptographic keys, as well as the encryption and decryption operations. The results of the measurements are based average time for the execution of 1 million operations for each protocol. GUIDs acted as input to hashing algorithms, while the produced hashes are used as an input to the asymmetric algorithms (just like it in the OT and PE protocols). The AES128 protocol is tested using a 1kB input (that is approx. 150 words of ASCII text) that is expected to be larger than necessary to simulate records returned by the dataholder (similar amounts of data used in [78] and [90]).

The test is conducted on a test machine running Microsoft Windows XP Professional with an AMD Turion 64 X2 Mobile 1.58GHz CPU, and 3GB of RAM. The results are provided in Table 4-1. From this comparison table it can be gathered that operations such as hashing and AES key generations are performed almost at wire-speeds and can be safely considered as negligible in this consideration. The large difference between key generation times for different asymmetric protocols can be explained by the fact that RSA generates a new pair of primes \(p\) and \(q\) each time, while in PH and SRA, the primes are common between the parties, and therefore are
generated only once per protocol run, or are part of the system. RSA has the smallest encryption time when compared to other asymmetric protocols examined, since most RSA keys use a default encryption exponent $e$ that is reasonably small (such as 0x10001) in order to speed up the encryption and signature verification processes, but not too small as not to expose the protocols to attacks described in [124]. On the other hand, the SHA and PH implementations cannot use such common choices for the encryption exponent, since these protocols expose the primes (used to calculate the decryption exponent) to the other parties, and thus the decryption exponent can be easily derived from the encryption exponent. PH and SRA use larger randomly generated numbers as the encryption exponents, and this result in longer encryption times.

<table>
<thead>
<tr>
<th>Conditions (bits)</th>
<th>Results (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>key generation</td>
</tr>
<tr>
<td>RSA 1024</td>
<td>723,344</td>
</tr>
<tr>
<td>SRA 7</td>
<td>7,128</td>
</tr>
<tr>
<td>PH 256</td>
<td>7,250</td>
</tr>
<tr>
<td>SHA 256</td>
<td>-</td>
</tr>
<tr>
<td>AES 128</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>1k</td>
</tr>
</tbody>
</table>

Table 4-1 Cryptographic operation performance measurements in nanoseconds (ns)

Table 4-2 shows the complexity of OT-based Approach 1 in an arithmetical format, while the complexity of the PE-based Approach 2 is shown in Table 4-3. The simplicity of operation of the OT-based approach contributes greatly to its high computational complexity. Since RSA is used as a trapdoor function of the OT, each record that needs hiding requires a separate RSA key. Key generation times for RSA are large in comparison to other asymmetric algorithms discussed in this chapter, since new primes $p$ and $q$ are generated for each new key. This is compatible with the typical use of the RSA protocol since each user needs only two different sets of keys (a separate set for signing and encryption). On the other hand, the PE-based approach only uses three asymmetric encryption (commutative PH) keys through the protocol, so the preparation phase for the protocol run is almost negligible. The computational complexity tables have been used to plot graphs illustrating the performance differences between the two approaches. Figure 4-5 depicts the total running time for both OT- and PE-based approaches including the preparation time, which is the time used to perform operations that are independent from the enquiry and can be performed prior to the protocol execution. The logarithmic graph shows that the total
running time for the OT-based solution is more than a magnitude higher than this characteristic for the PE based protocol, the exact values calculated are available in Appendix C. Figure 4-6 illustrates that when then preparation time is eliminated the performance of both protocols for $m = 1$ is of the same magnitude, with PE-based solution taking on average two thirds of the time used by the OT-base solution. It is worth noting that for both approaches the run-time is almost linear for the varying size of the dataset.

PE uses commutative encryption that employed in a 3Pass-like protocol adds similar benefits to $m$-$n$ SPIR protocols as public-key encryption had on securely exchanging information with multiple parties. Prior to emergence of public-key cryptosystems a party wanting to communicate securely with $n$ other parties would need $n$ different symmetric keys. Likewise, in order to achieve $m$-$n$ SPIR using encryption other than commutative $m$ different runs of 1-$n$ SPIR would often be necessary, while with commutative encryption and systems like PE as little as a single additional encryption operations is necessary to retrieve one more record. This is evident looking at the above complexity tables. In the OT-based approach in Step 3 of the OT phase the sender needs to decrypt the ciphertext received from the chooser $O(m \times n)$ times and then encrypt all $n$ records $m$-times in Step 4 resulting in $O(m \times n)$ complexity of symmetric encryption operations. The equivalent operations in the PE-based solution include the generation of the symmetric keys by encrypting (using the PH cipher) the hashed record ID and using the hashed result to encrypt the records using AES. Consequently, the equivalent operations in the PE require only $O(n)$ operations, each. Figure 4-7 presents this difference between the two approaches. Still, both protocols need to be praised for the use of symmetric encryption in hiding the data and using asymmetric ciphers to selectively transfer the symmetric keys between the parties. This is an optimal technique inspired by the PKI, and praised by Shundong et. al [105].

As mentioned already the OT protocol, used in Approach 1, is a simplistic protocol that is useful for illustrating the process of data acquisition. Its extensive preparation step requires generating $n$ different RSA keys, which makes it suboptimal for the requirements. However, the characteristic depicted in Figure 4-7 is similar to other OT protocols. Namely, OT protocols are usually optimised for handling a single
request from a large dataset per round of the protocol, hence the 1-n OT is the most common type of OT in use.

<table>
<thead>
<tr>
<th>PEqT</th>
<th>Symmetric Crypto. operation</th>
<th>SRA Cryptography</th>
<th>RSA Cryptography</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>key gen.</td>
<td>encrypt.</td>
<td>decrypt.</td>
</tr>
<tr>
<td>Preparation</td>
<td>-</td>
<td>O(2)</td>
<td>-</td>
</tr>
<tr>
<td>Step 1</td>
<td>-</td>
<td>-</td>
<td>O(m)</td>
</tr>
<tr>
<td>Step 2</td>
<td>-</td>
<td>-</td>
<td>O(m)</td>
</tr>
<tr>
<td>Step 3</td>
<td>-</td>
<td>-</td>
<td>O(m)</td>
</tr>
<tr>
<td>Step 4</td>
<td>-</td>
<td>-</td>
<td>O(n)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OT</th>
<th>Key gen.</th>
<th>Encrypt.</th>
<th>Decrypt.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 2</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 3</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 4</td>
<td>O(m × n)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 5</td>
<td>O(m)</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total Complexity</th>
<th>O(m(n + 1))</th>
<th>O(2)</th>
<th>O(2m + n)</th>
<th>O(m)</th>
<th>O(n)</th>
<th>O(m)</th>
<th>O(m × n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost (ms/operation)</td>
<td>0.564</td>
<td>7.128</td>
<td>21.125</td>
<td>51.297</td>
<td>723.344</td>
<td>0.766</td>
<td>47.266</td>
</tr>
</tbody>
</table>

Table 4-2 Computational complexity of the OT-based approach.

<table>
<thead>
<tr>
<th>PE</th>
<th>Symmetric Crypto. encryption</th>
<th>Asymmetric Crypto. encryption</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>key gen.</td>
<td>encrypt.</td>
</tr>
<tr>
<td>Step 1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 4</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 5</td>
<td>O(n)</td>
<td>-</td>
</tr>
<tr>
<td>Step 6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 7</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total Complexity</th>
<th>O(n)</th>
<th>O(m)</th>
<th>O(3)</th>
<th>O(3m + 2n)</th>
<th>O(2m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost (ms/operation)</td>
<td>0.551</td>
<td>0.564</td>
<td>7.250</td>
<td>22.389</td>
<td>50.594</td>
</tr>
</tbody>
</table>

Table 4-3 Computational complexity of the PE-based approach.

The literature review has identified the PE protocol as the only protocol that is optimised for retrieval (in a single round) of \( m \) records from a dataset. Since the data acquisition process calls for a solution that allows for retrieving of multiple records per enquiry (Req. 1) the PE protocol is the most likely choice to satisfy this condition. The load is almost linear to the number of records in the database. Since each record can be processed independently, there are no technical limitations to processing any number of records that can be stored in database. Therefore, PE would most likely meet Req. 7. The fact that it is possible does not necessarily mean that it is feasible. It would take at least eight days to process 15 million records on a standalone PC with a specification similar to the one used to generate the test data.
Consequently, Req. 3 would not be met, as if more than one enquiry would be run on such PC, the system would take more than two weeks to provide the response.

Figure 4-5 Total running time for both approaches including preparation time. Plotted for \( n \) varying from 50 to 100 million, and constant \( m \) equal to unity.

Figure 4-6 Data Acquisition processing time excluding preparation time. Plotted for \( n \) varying from 50 to 100 million, and constant \( m \) equal to unity.
If the results achieved are compared to other, similar systems for which data is available, PE still looks to out-perform the competition. The hardware-based TC solution presented in [78] initially requires the shuffling of all records before any request can take place. Shuffling 15 million records of 1.5kB in size would take more than a year using the SCOP. While the results discussed above show eight days as the total time required for the data acquisition of one record on a dataset of the same size, they concern records that are 1kB in size. However, increasing the record size to 1.5kB in the PE-based solution, increases the processing time by less than two hours. The shuffle needs to be performed only every $\sqrt{n}$ requests, while the PE protocol requires full run each time. Nevertheless, under PE a single request could contain $\sqrt{n}$ records.

Research presented in [90] describes protocols similar in functionality to the suggested approaches to the data acquisition process. The empirical results presented there suggest that the therein-discussed protocols are more efficient than PE, however, this empirical evaluation does not specify the size of the data, nor does it include the preparation phase in the considerations. The table illustrating the complexity of the protocols described by Cristofaro et. al. shows that it is similar to this of PE for $m = 1$, while the shorter total processing times are due to the use of the C as programming language, and a more powerful test bed PC. However, just
like the OT-based solution, the total processing time increases in line with the increase of the number of interesting records, whilst it is almost constant for PE-based solution.

Both approaches as presented, without the use of any balancers, require only a few rounds of communication. Consequently, if the time given for an enquiry was the statutory 14 days, the data could be exchanged using physical media rather than over the Internet, thus eliminating any limitations for the record and dataset size.

4.7.3 Feedback from practitioners

Following the initial evaluation, a member of ACPO and ACPO in Scotland (ACPOS) has been approached in order to obtain qualitative feedback for the results and clarify the investigative process. Also, a research poster shown in Appendix E has been presented during the second SIPR Annual Conference. The Detective Superintendent (DS) that cannot be named as the interview was based on Chatham House Rule is a member of the UK Data Communications Group – which compromises members of ACPO (and ACPOS), HMRC, and representatives of different CSPs. The DS was very interested in the research and stated the following in respect to the drafted requirements and assumptions:

1) Currently, an enquiry for communication data needed in a case where life of an individual is endangered takes minimum of 30 minutes.

2) Police have direct access to subscriber data, such as name and address, for all major CSPs.

3) Collateral damage to a data subject should be minimal as all enquiries are inspected by a designated person before being sent to dataholders, and are scrutinised in court of Law (if charges are pressed against the data-subject, or data acquired is used as evidence).

4) On some occasions, investigators need to postpone their enquiry, until they have enough background for the check (in order to protect data subjects against the collateral damage) or the subject is in custody (if there is suspicion that the data subjects may be informed about the enquiry taking place).
5) In face-to-face enquiries, law enforcement officers can ask general question allowing the person being interviewed to choose to amount of detail provided to the investigators. The technique is sometimes referred to as dilution. This is impossible in the current state-of-art in the digitalised enquiries, as it is often considered as fishing for evidence.

6) In some occasions, location data or other leads may need to be used to identify possible suspects and witnesses of an incident.

An example of a typical investigation is the use of evidence gathered from third parties given by the DS is the case of the Soham Murders (with accused being Ian Huntley), Maxine Carr provided an alibi for Huntley by stating he was with her at a specific location at the time of the murder. However, her phone location records obtained from her CSP showed she was 100 miles away. Therefore, the investigators could prove that she was lying. Newspaper articles confirm that the communication data was extensively used during the case, where the call timings are used to place individuals at different locations in a timeline of the events [132].

4.8 Requirements review

In light of the feedback from the practitioner and the empirical results, Req. 3 needs to be altered to reflect law enforcement expectations. Since, the minimum time for the complete enquiry is 30 minutes the protocols must be able to provide results even from large databases in less this time. However, as preparation time that takes place before, the enquiry can be permitted. The update is:

**Req. 3** Allow for efficient and timely data retrieval. (The protocol run excluding preparation should take less time than 30 minutes that it currently takes investigators to obtain investigative data in emergencies.)

Additionally, since different clues may need to be used to identify a potential suspect or a witness the data acquisition process should allow for a complex private matching criteria, that would allow selection of the records based on more than one column of data, and possibly allow for fuzzy matching. Thus, the following requirement should be added:
Req. 8 Provide a technique for multiple selection criteria of interesting records, and allow for fuzzy matching on the selection criteria different than record ID.

4.9 Conclusions

This chapter has identified and refined requirements for the data acquisition process. While some of these requirements are technical, such as the expected performance, other relate to the legal and social aspects of the process. These requirements were used to select a suitable PET primitive needed to facilitate privacy-preserving investigative data acquisition platform, and are used later in this thesis to evaluate the platform itself.

The PET primitives identified in the Literature Review have been scrutinised, and PE primitive allowing for private retrieval of records forming an intersection between two sets (in this case the set of potential suspects, and the set of all data subjects in a database) was chosen as a suitable protocol for the task-at-hand. In comparison to other protocols PE, run-time is almost independent from a number of suspects in an enquiry, while other protocols show almost linear increase in the total run-time with an increase in $m$.

Despite the PE-based approach being the best performing, it does not meet all the requirements (neither does any other considered protocol). Whilst it is capable of processing datasets of any size, as the records are processes one at the time independently from each other, it would take eight days to perform a retrieval on a database of 15 million using PE-based solution implemented in a managed C# .NET code and run on a computer similar in specification to the test setup. This could be improved with the use of different programming languages for the implementation and fast exponentiation, but still an enquiry would not complete in 30 minute as necessary (30 minutes is the current minimum time taken for an enquiry).

It is interesting whether such a solution could gain the acceptance of the general public. By choosing PE, a protocol based on commutative encryption that is relatively easy to explain, it would be likely to gain approval from the decision makers, as they already understand basic encryption terminology. However, the system sends all the records in the database to the chooser, using encryption to hide
the unselected records from the authorities, and members of the general public suspect public authorities of having computational power to break cryptosystems. For this reason, some additional measures should be built into the platform, in order to ensure that the chooser can prove that the data irrelevant to investigation has not been decrypted.

Another functionality that PE does not seem to provide is handling multiple selection criteria to identify interesting records. For example, if law enforcement officers are looking for a white female in her twenties, they cannot make such an enquiry privately against a corporate HR databases. It would be ideal if such cases could also be catered for, enhancing the authorities ability to identify potential suspects.
Chapter 5

Novel Data Acquisition Framework

5.1 Introduction

DAP is formed by improving on the shortcomings that PE primitive has in an investigative scenario. These shortcomings have been derived from the initial evaluation presented in Chapter 4 and include:

- Long processing times.
- Lack of capability to retrieve records matched on multiple selection criteria.
- Potentially low acceptance of the SPIR-based techniques.

The improvements that aim to addressing these shortcomings introduce a *dilution factor*, which is a numeric value that specifies the level of anonymity required for a given investigation. With this factor the data subject behind the interesting record
should feel assured that a constant level of privacy is provided to all individuals independently from the number of interesting records in an investigation.

The chapter presents a technique for forming complex privacy-preserving queries, without affecting the complexity of the protocol. This relies on joint hashing of the different selection criteria together, and using these as an input to the PEqT protocols. In order to gain approval of the general public a semi-trusted proxy is added as a novelty, in order to ensure information theoretic privacy of data-subjects whose records are not defined as interesting. Quantitative and qualitative evaluation of the complete approach to investigative data acquisition is planned and test implementation of IDAP is implemented.

5.2 Methodology

Chapter 4 has defined the requirements for data acquisition process and has shown that an information retrieval system based on PE primitive would be capable of meeting most of those requirements. PE is possibly the only information retrieval PET protocol that has almost constant processing time for enquiries, with a varying number of interesting records \( m \). This suggests that it is likely to be the most efficient \( m-n \) SPIR primitive. Still, processing of 15,000,000 records would take 8 days on the test bed used in experiments presented in Chapter 4. This could be shortened to less than 14 hours if the program is written in C (or C++) and run on a host similar to the one used in producing results for [90]. Thus, there is a clear need to improve the performance if the data acquisition process is going to employ the PE primitive. Other drawbacks of using SPIR-based PETs in obtaining investigative data are the lack of explicit functionality to retrieve records based of multiple selection criteria, and possible low levels of public acceptation (or understanding) for the SPIR concept, as it requires transferring data unrelated to an enquiry.

In order to design and implement IDAP, the shortcomings of the PE-based solution for data acquisition identified in Chapter 4 need to be mitigated. The modifications proposed in this thesis are based on the results of the initial evaluation and inspired by controls used in other PET primitives. The complete IDAP system is then defined and evaluated. The performance and the security of IDAP are discussed against the PE protocols and the requirements. This provides the pure quantitative evaluation of
the platform, while, in order to gain understanding of the public attitudes towards the protocol, a survey was carried out among IT security and privacy experts. Results from both experiments and survey are presented and discussed in Chapter 6.

5.3 IDAP Design

IDAP is formed on the basis of the PE primitive extended to fulfil the requirements outlined in Chapter 4. There are three modifications to this primitive that are required in order to facilitate the requests for investigative data. The resulting IDAP is a novel privacy-preserving approach to the data acquisition process.

5.3.1 Lowering Processing Time

There is a clear need to minimise the processing time required for each run of the protocol in large databases, such as those belonging to ISPs and mobile telephony providers. Theoretically, in order to maintain the privacy of the suspects, the sender needs to process all the records in the database per enquiry. Only in this way no information about interesting record is revealed and the correctness of the PE scheme can be proven under the rules of MPC [133]. Thus, if the data acquisition platform would use the PE primitive without modifications, the system would not be capable of processing any urgent requests due to the run-time required per enquiry, and this would be a major drawback. A possible mitigation against this could be to limit the numbers of records that are processed and sent by the sender per enquiry. This would also lower the communicational complexity that has not yet been taken into consideration in this thesis.

Privacy of the alleged suspect should be protected, but if the probability of the sender guessing the ID of the interesting record is for example 1:100,000 and not 1:n (for n being the size of the population or a large dataset), and the dataholder has no other information that could help infer the identity of the suspect, this research argues that the privacy of the suspect and the investigation is maintained. On occasion during traditional, i.e. face-to-face, information gathering exercises, Police Officers would use a concept of dilution – hiding the suspect’s identity by asking open-ended questions about a larger group of individuals rather than about a single person. This is a widely accepted technique, however, in a digital environment it is impossible to
build a system that would maintain privacy, while providing answers to such general questions. Consequently, any attempts of investigators to cast their net wide during electronic investigations are prohibited and treated as fishing-for-evidence. Taking in consideration that using the PET-based system the investigators will not get more data than required for their inquiry, limiting the set of records that are processed per enquiry should be acceptable. In comparison to methods used by the Internet users to protect their identities, there are estimate 100 000 active TOR clients any point in time and 1,500 TOR relays [82]. Thus, at best TOR users can expect to have only 1:100,000 privacy ratio.

The problem is to decide on the technique of narrowing down the scope in a way that ensures interesting records are among the results returned. If the list of the record identifiers is public, such as the list of the IP addresses or telephone numbers served by a given network operator, the chooser could simply select a number of random records from such directory in order to hide the true target of the investigation. Possibly, the chooser would first need to obtain a list of unused addresses from the provider, or at least know the percentage of unused addresses, in order to ensure that the number of unused addresses accidently included in the request does not reduce the level of privacy. However, in case of when a list of IDs is not publicly available, it would be possible to split the PE protocol back into separate parts: PEqT; and SPIR. In this way, the PEqT can be used during an initial preparation phase run against the whole dataset, and that the information retrieval would be performed against a smaller set of records.

It was previously mentioned that PE has almost constant processing time for enquiries with increasing number of interesting records (for low $m$ as shown in Figure 4-7). However, if the number of records retrieved per enquiry is lower than the size of the dataset it would be ideal if there is a constant level of privacy provided to each potential suspect. In the data mining field, there are already $k$-anonymity models that ensure that any privacy-protected statistical data record links to at least $k$ different identities [97]. Consequently, providing controlled level of privacy to the data-subjects. Relating to the concept of dilution used by the Police, a number of records requested per each interesting record can be defined as the dilution factor – $o$. This factor could be changed before each protocol run in order to allow
investigators to dynamically choose the appropriate level of protection for the given investigation, the data subject, and the data controller.

The proposed improved PE protocol operates by creating a single encrypted table of identities and allowing the investigators to privately match (using PEqT primitive) the identities of their suspects against this table. As the outcome of the private match operation the chooser would find out encrypted IDs of the interesting records. Then to perform an investigation the chooser would select \((o - 1)\) records at random per each interesting record from the encrypted table of IDs. The double encrypted IDs of the selected records would be communicated to the sender and remaining operations of the PE protocol would be run only on the selected records. Thus, the total number of requested records would be a product of the number of interesting records and the dilution factor, \((m \times o)\).

The described technique would introduce the potential for few different data controllers to collaborate and possibly identify the records of interest by checking for overlaps (intersection) of the requests made by the investigators to the collaborating data controllers. However, in the cases when the data is being retrieved from large databases that require use of the dilution technique during data retrieval process, the interesting records would usually be identified by a mobile phone number, or an IP address. Phone numbers and IP addresses are thus unique to the operators and their assignment can be obtained from call and network routing tables, respectively. Consequently, in most cases, the investigators would only need to ask a single operator for information about a given identity, and there would be no intersections of the requests. This fact makes most investigations equivalent to a single database SPIR allowing for dilution to be applied, with no adverse affect on the privacy of the data-subjects. The description of the improved protocol is as Figure 5-1, Figure 5-2 and Figure 5-3.

In this improved protocol the initial processing depends on the size of the dataset \(n\), but it needs to be performed only once in a given period of time. However, the remaining operations are run on limited dataset. Figure 5-4 illustrates the processes taking place in this improved version of PE protocol.
Phase A – Preparation:
1. **Sender** applies hash function $h$ to the elements in the input set $V_S$, so that $X_S = h(V_S)$.
2. **Sender** picks an encryption PH key $E_S$ at random from a group $Z'_p$, where $p$ is a strong prime.
3. **Sender** encrypts each $h(v) \in X_S$ with the key $E_S$, the result is a list of encrypted identities $Y_S = E_S(X_S) = E_S(h(V_S))$
   If more record needs to be added to the set these can be processes using steps 1 and 3, and then added to the list.

Phase B – Searching:
1. Following a request for data, **sender** provides **chooser** with a complete list of encrypted identities prepared during Phase A, reordered lexicographically.
2. **Chooser** applies hash function $h$ to the elements in set containing the identities of the interesting records, so that $X_C = h(V_C)$.
3. **Chooser** picks a commutative cryptography key pair, encryption key $E_C$ and decryption key $D_C$, at random from the same group $Z'_p$ that was used by sender in the Phase A.
4. **Chooser** encrypts entries in the set $X_C$, so that $Y_C = E_C(X_C) = E_C(h(V_C))$.
5. **Chooser** sends to **sender** set $Y_C$ reordered lexicographically.
6. **Sender** encrypts with key $E_S$ each entry $y \in Y_C$ received from **chooser**.
7. **Sender** returns set of pairs $(y; E_S(y))$ to **chooser**.
8. **Chooser** decrypts each entry in $E_S(Y_C)$ obtaining $E_S(X_C) = DCESEC(XC) = DCESYC$.
9. **Chooser** compares each entry in $E_S(X_C)$ to the entries of $Y_S$ received in the Step B1 (Step 1 of Phase B). This way the interesting records can be identified.
Phase C – Retrieval:

1. After identifying the interesting records in $Y_S$ the chooser selects at random $o-1$ other unique records from $Y_S$ for each interesting record in $V_C$. These are the diluting records, that together with the records of interest form a shortlist for the enquiry. If the number of interesting records multiplied by $o$ is greater than $n$, the size of the dataset $Y_S$, then the complete $Y_S$ is shortlisted.

2. Send the shortlist to sender.

3. Sender picks an encryption PH key $E'_S$ at random from the group $Z_p^*$.

4. Sender identifies entries $h(v)$ from $X_S$ that have been shortlisted and processes each shortlisted record in the following way:
   (a) Encrypts $h(v)$ with $E'_S$ to form the key used to lock the extra information about $v$, i.e. $ext(v), \kappa(v) = E'_S(h(v))$.
   (b) Encrypts the extra information using a symmetric encryption function $K$ and the key $\kappa(v)$ crafted in the previous step: $c(v) = K(\kappa(v), ext(v))$
   (c) Forms a pair $(E_S(h(v)), c(v))$.

5. The pairs formed in C4(c), containing a private match element and the encrypted extra information about record $v$, are then transferred to chooser.

6. Sender encrypts each entry $y \in Y_C$, received from chooser in Step B5, with key $E'_S$ to form set of pairs $(y; E'_S(y))$.

7. Pairs $(y; E'_S(y))$ are then transferred to chooser.

8. Chooser removes the encryption $E_C$ from all entries in the 2-tuples received in Step C7 obtaining tuples $\alpha, \beta$ such that $(\alpha; \beta) = (h(v); E'_S(h(v)))$. Thus, $\alpha$ is the hashed value $v \in V_C$, and $\beta$ is the hashed value $v$ encrypted using $E'_S$.

9. Chooser sets aside all pairs received in Step C5, whose first entry is equal to one of the first entry of any
two-tuples obtained in Step B9. Then uses the appropriate $\beta$ tuple associated with a given interesting record as a symmetric key to decrypt the extra information contained in the second entry in the pair received in C5. This is performed for all the matching entries.

Figure 5-3 Lowering Processing Time Phase C – Retrieval

Figure 5-4 Process flow of the protocol incorporating the dilution factor
5.3.2 Allow multiple selection criteria

The PE protocol can be used to privately retrieve data if the data is identified by a single parameter, such as ID number, credit card number, IP address, and so on. However, this is not always the case. If data acquisition process is used to find a suspect based on circumstantial knowledge, or a suspect’s profile, the PE protocol would need to be modified. The query shown in Figure 5-5 shows the way the request from Figure 4-2 would be modified for such enquiry, here $sip_{1:j}$ stand for $j$ secret input parameters (sip):

```
SELECT sip_1, sip_2, ..., sip_j, rp_1, rp_2, ..., rp_H
FROM source
WHERE ip_1 = ip_val_1 AND ... AND ip_L = ip_val_L
```

Figure 5-5 mapped into SQL

A computationally expensive solution to this problem can be achieved by using symmetric encryption to lock the return parameters and then hiding the symmetric keys used with the commutative encryption keys unique to each value of the secret input parameter. The chooser would then perform a separate PE-based retrieval of the asymmetric key for each interesting value of the secret parameters (such as age equal to 25 years). Since these asymmetric are commutative, the chooser would be able to decrypt the ciphertext containing the symmetric key that was used to lock records matching the selection criteria. Despite being computationally-expensive this solution has a unique benefit of allowing semi-fuzzy matching of the results if the underlying commutative protocol is ElGamal-based. This is the case as ElGamal (and its commutative form suggested by Weis in [57]) uses checksums that allow for verifying whether a given ciphertext can be decrypted with a given key. Thus, it would possible to establish how many records match each secret input parameter. This solution has been published in [13], however, it is not suitable for large databases due to its high computational complexity.

In this thesis a simplified approach is proposed. Since, the query from Figure 5-5 replaces the $ri$ parameter with $j$ different $sip$ parameters then the list of these $j$ parameters could be used as a complex $ri$ for use with PE-based data acquisition protocol. Thus, in steps B2 and A1 of the protocol presented in Section 5.3.1 a list of
all values of given \( sip \) parameters would be hashed together to form records in sets \( V_C \) and \( V_S \). In this way neither the security, nor the complexity of the protocol is affected by this improvement (if processing time required to produce hashed values is considered to be negligible).

### 5.3.3 Reassuring the Public

The initial design of IDAP proposed in the form of two approaches to data acquisition process investigated in Chapter 4 would shift the balance of the privacy protection from innocent individuals towards the suspect and the secrecy of investigation. Currently, the data acquisition process employed by the public authorities does not affect privacy of the data-subjects whose records are not of interest to the investigators, as there is no need to process these records. IDAP changes this as per each enquiry there is a number of records unrelated to the investigation returned to the chooser. The fact that the chooser is unable to decrypt these records does not change the fact that the records are being processed (according to the DPA definition of processing). As the anonymity in the PE protocol is based on hiding the interesting records among other records, some records unrelated to the investigation will always be retrieved by the chooser. Thus, there is a need to ensure that the chooser does not abuse the system. As Juvenal put it:

\[ \textit{Sad quis custodiet ipsos custodies?} \quad (\textit{Juvenal, Satires VI, 347}) \]

Which translates to: \textit{But who will watch the watchers?}

It is likely that providing government agencies with records of innocent, individuals unrelated to any investigation would worry the general public. This is despite the data being encrypted in the way that renders these records unusable to the authorities i.e. secure against attacks in polynomial time. However, the public may worry that government organisations have enough computing power to break the encryption used in IDAP. There are few actions that may reassure the public that the data is safe. First, if the technique for minimising the processing time presented in Section 5.3.1 is employed, the chances that investigators will retrieve encrypted records of a particular individual that is not a suspect is small in large datasets. Thus, for a dataset
with $n$ records, during investigation with $m$ interesting records and the dilution factor $o$, the probability of this event $A$ can be defined as:

$$P(A) = \frac{(o-1)\times m}{n-m}$$

Eqn. 5-1

Consequently, for investigation with five interesting records, with dilution factor of a 1,000 and dataset consisting of a 15 million records, the probability of this event occurring during a single run of the protocol would around 3%. This also means that the investigators would need to first break the encryption key used by the sender to hide identities (Phase A), before they could attempt to obtain the data about a specific individual that is not a suspect, otherwise the probability of the encrypted record for this individual being provided to them would be small. Thus, if the identity of a data subject were never encrypted under the same key as the data records then investigators would need to successfully brute force two separate keys in order to link any record not declared as interesting to an identifiable individual. Otherwise, the information would be unintelligible, or random.

The merits of the above discussion could certainly improve the perception of the system. Still, currently there is greater trust in security processes than encryption, as inferred from [91, 134]. The solution proposed in this thesis in order to reassure the public is to introduce a semi-trusted party into the protocol. Such a party is often used in PET protocols in order to balance the computational and communicational complexity between the participating parties or off-load processing from the participants [113]. This party would become a proxy between the investigators and the dataholder, however, unlike in other PETs, the purpose of this proxy is to ensure that investigators get only the interesting records and all other records sent to them during the PE protocol are discarded, thus the solution to gain public’s trust is proposed in Figure 5-6.

The semi-trusted party should thus have no interest in finding out the object of the investigation (the interesting record ID) or the content of the data records returned by the dataholder. For this reason, it is suggested that the role of this party should be conducted by an independent body trusted by the public. In the UK for example, the Information Commissioner’s Office (ICO) is such an independent body that ensures
DPA and RIPA are adhered to. Thus, the ICO would be an ideal organisation to become the *proxy* and could help restore the natural order, where the rights of the innocent are put ahead of the secrecy of the investigation.

1. All communication between *chooser* and *sender* goes through *proxy*.
2. Chooser provides *proxy* with the identifiers of the interesting records encrypted by *sender*, $E_S(h(v))$. This is done over a secure channel or with use of a 3Pass protocol once the parties are authenticated.
3. At the stage where data is transferred from *sender* in Step C4, *proxy* filters the response and discards the records that were not specified by *chooser’s* request, this is the records other than the ones identified in Step 2.

**Figure 5-6 Reassuring the public by introducing semi-trusted third parties**

It must be noted that the party that is chosen to become the *proxy* must not cooperate with the *sender* or the protocol will be broken, since simple matching exercise would reveal the identities of the interesting records (but not the data). A key concept is that the *proxy* has no incentives to find out the detail of the investigation so it is going to cooperate with the *sender* in order to establish the identity of the suspect. On the other hand, if the need arises to verify the *chooser’s* requests in front of a court of law, the *proxy* and the *sender* could work together to establish the identities of the records requested by the *chooser*.

In law if a party refuses to provide the evidence needed by another party a commissioner can be appointed to gather evidence listed on the Specification of Documents prepared by the requesting party. This commissioner then verifies whether the requesting party needs any given piece of potential evidence, and provides this party with only the documents relevant to the investigation. The process is referred to as *commission and diligence* [31, 32], and this is the legal justification for the introduction of the semi-trusted *proxy* into the data acquisition process.
5.4 Implementation

For evaluation purposes only, IDAP was implemented using C#.NET programming language allowing it to run on Microsoft Windows NT 5.0 and higher platforms. However, the design is not specific to a programming language. In fact, as it is possible to implement the symmetric encryption and hashing with no modifications to known cryptographic programming suites and only small modifications are required to implement PH encryption, the system can be deployed using virtually any popular programming language onto any operating system and hardware that supports RSA encryption.

The implementation is performed in order to facilitate the discussion of the platforms performance. Although, it is possible to compare the platform to other PET protocols using their theoretical complexity, this is insufficient to evaluate the platform against the requirements. The running time can be estimated using the average cryptographic operation times provided in Table 4-1, but there is no certainty that the model used to produce these estimates is not too simplistic. There is possibility that the operations considered as negligible, in terms of the required processing, by the model in the implemented solution do affect the performance. Therefore, IDAP is implemented in order to confirm the results achieved from the model. In this way, if the modelled complexity gives a good illustration of the results then it will suffice as a source of data for other experiments.

Section 5.3 has provided the overall design of IDAP, which has highlighted that IDAP is only a tool that should be used within the well-vetted data acquisition processes already used by the public authorities, and other public authorities. Figure 5-7 depicts the way that IDAP fits into the data acquisition process. In this figure blue arrows signify requests, which between the chooser and the sender include randomly generated \((m \times o)\) identities, while the response from the sender is filtered by the proxy to provide the chooser with \(m\) interesting records, only.

Implementation of the encryption protocols has already been discussed in Section 4.7.1. It is possible to implement these with only small changes to common open source cryptographic suites. Most OT and PIR primitives are evaluated on a single machine, since the communicational complexity can be easily modelled and there is
no real need to run processes in parallel during testing. However, in order to establish whether this test methodology is correct, the implementation of IDAP makes it a distributed application. For this reason, data must be transferred between the parties in this way that is most optimal for the large number of records being transferred. In the Microsoft .NET framework, data can generally be transferred in the raw binary format or using XML. The first method is usually preferred for large files, since the raw format is thought to contain less overhead and does not need to be specially encoded for transport. However, the first is true only for certain binary data items such as pictures, videos and sound clips, since even binary files contain metadata describing the file type and structure. The more complex the structure, the more metadata is required. On the other hand, in ordinary XML files, binary data (such as the encrypted records) needs to be encoded, which can results in an unnecessary overhead. As an example, the encoding-overhead ratio is 4:3 if binary data is Base-64 encoded for transport. Finally, Simple Object Access Protocol (SOAP) technology called Message Transmission Optimization Mechanism (MTOM) can be used to embed raw binary data within XML. This combines the benefits of the raw format and XML structure for objects in cases where the size of records is higher that 1kB [135]. Consequently, MTOM is employed in data exchanges between the parties.

Figure 5-7 IDAP
Common programming practice is to receive the whole piece of data into a buffer, when transferring it over a network. This technique, although perfectly valid, is not suitable for larger pieces of data and streaming should be used instead for large chunks of data. However, in case of IDAP, the data is small in size but large in values so streaming is not necessary, and the data should be sent across the network in relatively small messages, as to enable buffering of these messages [135].

The control messaging between the three different players in the system (chooser, proxy and sender) is handled by the Windows Communication Foundation where the chooser is the client of services, provided by the proxy, and the proxy is the client of the sender.

5.5 Proposed Quantitative Evaluation

The quantitative evaluation of IDAP assesses the validity of the customisations to the PE primitive discussed in Section 5.3. Results from the evaluation are provided and discussed in Chapter 6.

5.5.1 Overall design of experimental environment

IDAP consists of three applications: chooser (client); proxy (client and server); and sender (server). The performance of each of these applications needs to be evaluated, however, it should be noted that due to the way the platform has been designed the proxy does not need to process records, as it simply relays messages between the choosers and the senders, and filters the results returned from data acquisition queries. The major metrics that are necessary to evaluate the protocol are:

- Processing time per operation.
- Bandwidth used.

However, in order to establish the strain that IDAP puts on hardware a number of secondary metrics needs to be collected during the experiments:

- CPU usage.
- Memory usage.
The method used to collect these data cannot be resource intensive, as not to affect the results. For this reason, the processing time is measured by taking a timestamp before and after an operation, and the time frames for each operation are calculated and stored into results database at the end of the protocol run. An example, shown in Figure 5-8, demonstrates measuring processing time for symmetric encryption performance test.

```csharp
byte[] messageBytes, outputBytes;
Hashtable encrypted = new Hashtable();
start = DateTime.Now;
for (int i = 0; i < count; i++)
{
    messageBytes = (byte[])Encoding.Unicode.GetBytes(input);
    outputBytes = symHlpr.performEncrypt((byte[])keys[i], messageBytes);
    encrypted.Add(i, outputBytes);
}
end = DateTime.Now;
step_2 = Convert.ToInt32(((TimeSpan)(end-start)).TotalMilliseconds);
```

Figure 5-8 Measuring processing time

Bandwidth used, memory and CPU load are measured in a similar manner by an external application. Prior to the system entering a given stage in the program the performance measurement subroutine is run that reads the number of the bytes transmitted and received on a given networking adapter (Figure 5-9), then the probing of the memory and CPU usage takes place. Once the given stage of the IDAP program is over, the subroutine is terminated, statistics for the network adapter read again, the used bandwidth, as well as average memory and CPU usage calculated.

```csharp
string networkCard = "";
static PerformanceCounter dataSent;
static PerformanceCounter dataReceived;
ArrayList bandwidthSamples = new ArrayList();

public static KeyValuePair<float, float> GetNetworkUtilizationNow()
{
    return new KeyValuePair<float, float>(dataSent.NextValue(),
                                           dataReceived.NextValue());
}
```
private void initialiseNetCounters()
{
    dataSent = new PerformanceCounter("Network Interface", "Bytes Sent/sec",
         networkCard);
    dataSent.NextValue();
    dataReceived = new PerformanceCounter("Network Interface", "Bytes Received/sec",
         networkCard);
    dataReceived.NextValue();
}

public void client_StartFullEnquiryCompleted(object sender,
    StartFullEnquiryCompletedEventArgs e)
{
    ...
    initialiseNetCounters();
    bandwidthSamples.Add(GetNetworkUtilizationNow());
    ...
}

private void decryptData()
{
    ...
    KeyValuePair<float, float> full_stop_band = GetNetworkUtilizationNow();
    bandwidthSamples.Add(full_stop_band);
    KeyValuePair<float, float> full_start_band = (KeyValuePair<float,float>)
        bandwidthSamples[0];
    float totalSent = (full_stop_band.Key - full_start_band.Key);
    float totalReceived = (full_stop_band.Value - full_start_band.Value);
    ...
}

Figure 5-9 Measuring bandwidth used during a protocol run

5.5.2 Experiments

The experiments forming the quantitative evaluation of IDAP mainly consist of performance measurements in different scenarios. However, the initial experiment assesses whether modelled complexity of IDAP is suitable source of data for evaluation. This is achieved by comparing the results obtained from a model of IDAP with measurements taken from scenarios run on the implementation of IDAP.

Evaluation of the complexity model

The main objective of this experiment is to check how closely modelled complexity matches performance of the implementation, which should reflect on the programming language and run environment used for the implementation. Thus, for
simplicity, the experiment is conducted using PE implementation and not IDAP. Both are made with the same components, so results achieved will apply to IDAP, just as well.

**Varying the number of interesting records**

Some proposed modification to PE set out to improve significantly the processing time required for each protocol run. This experiment run on PE and IDAP is used to compare the performance of these protocols for different queries with varied number of interesting records requested. It has already been shown that processing time for PE is almost constant for increasing number of interesting records. On the other hand, in the graph outlining the performance of the PE in similar experiment (Figure 4-7) it can be seen that for high number of interesting records \( m \) the processing time is no longer linear, and grows almost exponentially. This experiment will be run a number of times for different values of \( n \) in order to evaluate the maximum limit of interesting records per enquiry.

**Varying number of enquiries**

In IDAP there can be multiple parties acting as *choosers* and *senders*, but logically there can be only one *proxy*. Physically, the *proxy* can consist of a number of devices, but only one party can be put in the position of trust to oversee the data acquisition requests. Otherwise, it would be possible for a *chooser* to cheat and make a number of similar requests to the same *sender* via different proxies, potentially allowing the *chooser* to *fish for evidence*. Taking into consideration that there are many different public authorities that may require investigative data from third parties, and the major CSPs are the most likely targets of the DPA and RIPA notices the load on the *proxy* and the *sender* need to be evaluated for growing number of simultaneous enquires.

**Evaluation of IDAP with access to a directory of identities**

Many efficient SPIR solutions exist for scenarios where directory of records is public. For example, such SPIR techniques are proposed for privacy-preserving purchases [84]. While on the first sight it is unlikely for the investigators to have a list of the identities in the dataset, this is not always the case. The Police has already direct access to subscriber data of large CSPs, and HMRC knows the names (and
other details) of employees in an organisation. In this experiment performance of the IDAP using a dictionary of the dataset is going to be measured against performance of ordinary IDAP that runs the PEqT protocol against the complete dataset. This is going to be measured for varying number of enquiries and varying number of interesting records.

5.5.3 Proposed Qualitative Evaluation

It is possible to quantitatively evaluate the performance of IDAP; however, this thesis sets out to provide a complete solution to the problem of privacy in data acquisition. A problem that is not purely technical. An integral part of privacy evaluation is an assessment of perception of a given system. This is because privacy is different for each and every individual, and the legislations in UK and other European countries enforce this, by giving all individuals certain level of control over their data. For this reason, qualitative evaluation is necessary for IDAP. This major evaluation is carried out in a form of survey targeted at security and privacy experts. This evaluation is complemented by a discussion of IDAP’s security.

Survey

A website [136] has been set-up to host the survey. On the survey website the area of the research is explained and brief explanation of IDAP provided. After collecting some details relating to the participants’ interest in the subject, the introductory questions ask the participants about their (and their organisation’s) security practices, such as the use of secure communication and storage, and their attitude towards digitalisation of the data acquisition process. The remaining questions introduce the privacy problems in this process and propose solution, requesting quantitative and qualitative feedback from the participants. Thus, each single- and multiple-choice question is followed by a text box allowing the participants to express their answers, and opinions, in less rigid manner. Please see Appendix D for to see the survey questions.

Correctness and Security

Security of IDAP is verified against the requirements of MPC [90] and in contrast to the cryptanalytic attacks outlined in Section 2.5.6. The PE primitive has already a good proof of security, and this will be presented here based on [63], while the
discussion will focus on the effect that changes to PE introduced by IDAP influence security.

5.6 Conclusion

Building on the results presented in Chapter 4, where PE has been chosen as most likely primitive to facilitate privacy-preserving data acquisition process, necessary improvements to this primitive are proposed. It has been identified that despite PE being most likely the efficient protocol that can privately match and retrieve investigative data. However, retrieving records from large databases with (15 million and more records) is not viable. Consequently, a technique for narrowing down the scope of data retrieval is designed and implemented for evaluation.

Generally, when using privacy-preserving information retrieval techniques, it is only possible to match records on a single selection criterion, such as a record ID. In this thesis it is proposed that a number of different selection criteria can be combined together by linking the sought-after values in one string and hashing this string for use in the same way as a record ID. In this way, it is possible to privately search databases with minimal complexity increase. However, this technique may place considerable load on the database, thus an experiment is proposed to evaluate this load.

The third, and final improvement potentially needed by the PE primitive to become viable for data acquisition process is addition of semi-trusted third party. This thesis, and the whole field of MPC, is based on the real-life assumption that trusted third party does not exist in most cases. Looking at the currently deployed privacy-preserving technologies procedures are more trusted than encryption / technical measures Thus, a semi-trusted model where a party proxies the communication between the investigators and the dataholder, and ensures that the investigators do not receive any other data records than those specified as interesting, can benefit public’s trust. Such party, called proxy, would have to be mutually trusted by the public authorities, the dataholders and the data subjects. In UK ICO could potentially become the proxy for all data acquisition requests,
The improved PE primitive forms IDAP and then suitable evaluation techniques are discussed. This includes quantitative evaluation aiming to establish whether the performance of the IDAP is satisfactory, as well as qualitative evaluation seeking feedback from security and privacy specialists, as well as grading IDAP’s security based in available literature.
Chapter 6

Evaluation

6.1 Introduction

The PE protocol was selected as the most suitable basis for an investigative data acquisition protocol. However, Chapter 4 identified a number of shortcomings of this protocol in an investigative scenario that have been analysed in Chapter 5. This resulted in definition of a novel approach to data acquisition referred to as IDAP. This approach needs to be evaluated in terms of performance and correctness.

IDAP’s performance is, therefore, evaluated against the PE protocol. But first, the methodology of using simulations for assessing the performance of this type of protocol is put to test in an experiment where empirically-gathered performance data is compared to a simulation. This confirms the methodology of modelling complexity of a protocol in order to evaluate it, commonly used in this field, is correct. Finally, the conducted simulations clearly outline the benefits of using IDAP over PE.
Survey results are discussed and suggest that hiding the identities of alleged suspects is the correct solution to the problems of privacy and secrecy in the investigative data acquisition process. This shows that the main benefit of IDAP over the current processes is in-line with the expectations of privacy and security practitioners.

6.2 Presentation of performance impact

Performance of IDAP is put to test using experimentation and simulation. It is shown that IDAP performs better that PE under most circumstances.

6.2.1 Evaluation of the complexity model

The main purpose of this experiment is to establish whether it is possible to evaluate IDAP based on modelled complexity, rather than experimental results taken from the implementation of the complete platform. The literature suggests using the notion of complexity to compare different protocols, however, most often only the most costly operation is included in the considerations of computational complexity. Thus processes such as symmetric encryption and hashing are often ignored, and a number of asymmetric encryption operations is used to express computational complexity [90]. In a similar fashion, communicational complexity is often expressed as a function dependant mainly on number of records processed. In this thesis analogous, but more precise technique, is employed. Chapter 4 provided a complexity table for the PE protocol outlining the number of cryptographic operations required for the protocol run. These outlined operations included symmetric and asymmetric (commutative) cryptographic functions. Also, the cost of each operation evaluated empirically is expressed in milliseconds per operation. Such tables can potentially be used to plot performance graphs for the different uses of the protocols. However, some empirical testing needs to be performed in order to ensure that this simulation technique is fit-for-purpose. Thus, in this experiment the modelled complexity of PE is compared to results obtained from the C# .NET implementation of this protocol (outlined in Appendix B). Figure 6-1 illustrates this comparison in a simple scenario with 1,000 records in the sender's database and a varying number of interesting records. The tree lines in this graph represent:
• The total time required for a protocol run, derived from a summary computation complexity for the chooser and the server.

• The total time needed by the implementation to provide results. Measured from the callback `client_StartFullEnquiryCompleted` till the end of processing the records by the `decryptData` method (Appendix B).

• The summary of the time required by the chooser and the server to compute and to exchange the results. This is the sum of the run times for sections C1, C3 and S1 (Appendix B).

![Figure 6-1 Complexity table reading vs. actual measurements](image)

There is no direct link between the total time taken per enquiry and the total time achieved using the complexity tables. The run times are of the same magnitude, but the line illustrating the simulated run-time curves up with a lower count on the number of interesting records. This can be explained by the process flow of the implementation. By default the chooser performs calculations on the IDs of the interesting records, while the server prepares the dataset, by creating packages for each record according to the query received from the chooser. Thus, the some operations may run in parallel. However, this fact is not taken into account in the complexity table presented in Table 4-3 and used for this experiment. So, in fact, the calculated total run-time should be compared to the sum of time required by the chooser and the sender. In the graph, the curve of the line illustrating the measured
run-time matches closely the curve plotted based on the simulated results. Therefore, the complexity tables do reflect closely on the performance of the implemented protocol. However, the fact that some operations are conducted in parallel by two separate parties needs to be taken into account when discussing IDAP’s performance.

Taking into consideration these results, the complexity table for IDAP should allow for different ways to assess the processing times. Consequently, these should include separate definition of complexity for the chooser and for the sender. Table 6-1 provides a detailed breakdown of IDAP’s computational complexity. It includes the parameter \( k \) that expresses the number of enquiries subsequent to the periodical encryption of all IDs in the system performed by the sender (Phase A).

<table>
<thead>
<tr>
<th>Phase A (run periodically)</th>
<th>Symmetric Cryptography</th>
<th>Asymmetric Cryptography</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>-</td>
<td>( O(1) )</td>
</tr>
<tr>
<td>Step 3</td>
<td>-</td>
<td>( O(n) )</td>
</tr>
<tr>
<td>Phase B (run per enquiry)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>-</td>
<td>( O(1) )</td>
</tr>
<tr>
<td>Step 4</td>
<td>-</td>
<td>( O(m) )</td>
</tr>
<tr>
<td>Step 6</td>
<td>-</td>
<td>( O(m) )</td>
</tr>
<tr>
<td>Step 8</td>
<td>-</td>
<td>( O(m) )</td>
</tr>
<tr>
<td>Phase C (run per enquiry)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>-</td>
<td>( O(1) )</td>
</tr>
<tr>
<td>Step 4(a)</td>
<td>-</td>
<td>( O(m \times n) )</td>
</tr>
<tr>
<td>Step 4(b)</td>
<td>( O(m \times n) )</td>
<td>-</td>
</tr>
<tr>
<td>Step 6</td>
<td>-</td>
<td>( O(m) )</td>
</tr>
<tr>
<td>Step 8</td>
<td>-</td>
<td>( O(m) )</td>
</tr>
<tr>
<td>Step 9</td>
<td>-</td>
<td>( O(m) )</td>
</tr>
<tr>
<td>Total Complexity</td>
<td>( O(k \times m \times n) )</td>
<td>( O(k \times m) )</td>
</tr>
<tr>
<td>Cost (ms/operation)</td>
<td>0.551</td>
<td>0.564</td>
</tr>
</tbody>
</table>

Table 6-1 Initial definition of IDAP’s complexity.

This complexity table can be further refined. Symmetric encryption and decryption times are almost identical, 551\( \mu s \) and 564\( \mu s \) respectively, so the complexity of the operations can be summarised and the cost rounded-up to 0.6ms/operation. In IDAP there is no need for a large number of asymmetric cryptographic keys, as for a single run of protocol there are only three keys required, which means that asymmetric key generation should not affect the protocol run times. On the other hand, asymmetric encryption contributes the most towards the computational complexity. This can be observed in the Figure 6-2, as the processing time for asymmetric encryption is few magnitudes higher than any other component of the complexity table.
The presented complexity table, and the resulting graph, are imprecise, as they do not reflect all conditions. First, and foremost, if the product of the number of interesting records, and the dilution factor, is higher than the number of records in the dataset (this means for the cases where condition from Eqn. 6-1 is not met) the pure version of the PE protocol, as presented in [63], needs to be employed:

$$m \times o \leq n$$  

Eqn. 6-1

But such scenarios should be avoided in practice, as they cannot guarantee $k$-anonymity of the alleged suspects. Nevertheless, in order to evaluate IDAP’s processing, the complexity of PE is used for such cases. This is reflected in Figure 6-2.

<table>
<thead>
<tr>
<th>Condition</th>
<th>Symmetric Crypto. operation</th>
<th>Asymmetric Crypto</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Key gen.</td>
<td>Encryption</td>
</tr>
<tr>
<td>Total complexity</td>
<td>$O(m \times o \leq n)$</td>
<td>$O(k \times m \times (o + 1))$</td>
</tr>
<tr>
<td></td>
<td>$O(m \times o &gt; n)$</td>
<td>$O(k \times (n + m))$</td>
</tr>
<tr>
<td>Chooser’s complexity</td>
<td>$-$</td>
<td>$O(k \times m)$</td>
</tr>
<tr>
<td>Sender’s complexity</td>
<td>$O(m \times o \leq n)$</td>
<td>$O(k \times m \times o)$</td>
</tr>
<tr>
<td></td>
<td>$O(m \times o &gt; n)$</td>
<td>$O(k \times n)$</td>
</tr>
</tbody>
</table>

Table 6-2 IDAP’s complexity.

The second omission from the complexity show in Table 6-1 and Table 6-2 is the scenario where the chooser can obtain a directory of the entries in the dataset. For
example the Police have direct access to the subscriber data of large CSPs in UK, thus, they can use this access to narrow down the scope of the enquiry so that not the complete dataset of a large CSP needs to be encrypted. Also, the CSPs make public the ranges of the identifiers that they manage, such as IP addresses, or telephone numbers. Thus, with a certain probability, it is possible to generate an enquiry that ensures there are at least \( o \) different active identities (addresses) that can be linked to every interesting record. The computational complexity for this variation of the protocol is shown in Table 6-3.

<table>
<thead>
<tr>
<th></th>
<th>Condition</th>
<th>Symmetric Crypto. operation</th>
<th>Asymmetric Crypto</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Key gen.</td>
<td>Encryption</td>
</tr>
<tr>
<td>Total complexity</td>
<td>( O(m \times o \leq n) )</td>
<td>( O(k \times m \times (o+1)) )</td>
<td>( O(3k) )</td>
</tr>
<tr>
<td></td>
<td>( O(m \times o &gt; n) )</td>
<td>( O(k \times (n+m)) )</td>
<td>( O(3k) )</td>
</tr>
<tr>
<td>Chooser's complexity</td>
<td>-</td>
<td>( O(k \times m) )</td>
<td>( O(k) )</td>
</tr>
<tr>
<td>Sender's complexity</td>
<td>( O(m \times o \leq n) )</td>
<td>( O(k \times m \times o) )</td>
<td>( O(2 \times k) )</td>
</tr>
<tr>
<td></td>
<td>( O(m \times o &gt; n) )</td>
<td>( O(k \times n) )</td>
<td>( O(2 \times k) )</td>
</tr>
</tbody>
</table>

Table 6-3 IDAP's complexity for datasets with publicly available dictionaries.

### 6.2.2 Varying the number of interesting records

The purpose of this test is to evaluate IDAP’s performance in contrast to PE for varying number of interesting records. By customising PE to a specific application of investigative data acquisition IDAP has introduced a number of potential improvements to the efficiency of this information retrieval protocol.

Figure 6-3 illustrates processing time for IDAP and PE for varying number of interesting records (Eqn. 6-2). It can be seen that initial processing time for IDAP is slightly lower than the PE processing time, and, as designed, both are the same in the range where the product of \( m \) and the dilution factor \( o \) is higher than the size of the dataset \( n \) (Eqn. 6-3).

\[
1 < m < n \quad \text{Eqn. 6-2}
\]
\[
m \times o > n \quad \text{Eqn. 6-3}
\]

The graph shown in Figure 6-4 focuses on the range of values that do not meet condition in Eqn. 6-3. IDAP performs significantly better than PE for low values of \( m \), or, to be more precise, when ratio of \( m \) to \( n \) is small. This experiment needs to be
repeated for different values of $n$ and $m$ in order to establish the optimal value of the ratio of $m:n$, and the effect of the dilution factor on the optimal value of this ratio. The graphs that help establish this connection are presented in Figure 6-5 and Figure 6-6. Figure 6-5 shows that IDAP’s processing time is almost constant for low values of $m$ if other parameters ($n$ and $o$) are constant. The point where the line showing the processing time curves up represents the optimal operation of IDAP. However, Figure 6-6 shows that the processing time is also dependent in a similar, but inverse, way on $o$ the dilution factor. If the proportion of the three key parameters ($m$, $n$, and $o$) is referred to $\gamma$ it can be expressed as in Eqn. 6-4 (this follows from Eqn. 6-3). Thus, since the processing time is constant for $m \leq 100, o = 1000, n = 1000000$, and so on, the operation of IDAP is optimal for $\gamma = 0.1$ (Eqn. 6-5), if there is no overlap between the diluting records.

$$\gamma = \frac{m \times o}{n}$$  \hspace{1cm} \text{Eqn. 6-4}

$$\gamma = \frac{100 \times 1000}{1000000} = \frac{1}{10}$$  \hspace{1cm} \text{Eqn. 6-5}

Figure 6-3 Computational complexity of IDAP and PE for increasing $m$ (logarithmic scale)
Figure 6-4 Computational complexity of IDAP and PE for increasing $m$

Figure 6-5 IDAP’s processing time for varying $m$ and different values of $n$

Figure 6-6 IDAP’s processing time for varying $m$ and different values of $o$
6.2.3 Varying number of enquiries

This experiment focuses on the evaluation of IDAP, in contrast to PE, for scenarios with multiple enquiries against the same dataset. In IDAP $k$ different enquiries can be made against a single encrypted list of identities within a dataset. This would be typical scenario for organisations that often provide data to the public authorities, such as CSPs. Figure 6-7 and Figure 6-8 depicts the results of this experiment. The key point to take from these graphs is that under the test conditions ($n = 1\text{mln}; m = 10; \sigma = 1000$) the processing time for PE is two orders of magnitude higher than IDAP’s processing time. In addition, IDAP’s processing time is almost constant for $k \leq 10$.

The value of $\gamma$ affects the results and therefore the effect that the ratio defined in Eqn. 6-4 has on these results is further evaluated in Figure 6-9. The maximum value for $\gamma$ is one in environment where diluting records do not overlap, but even for $\gamma = 1$ IDAP performs significantly better than PE in a data acquisition scenario.

![Figure 6-7 Comparison of processing time for IDAP and PE. For varying $k$, $n=1\text{mln}$, $m=10$, and $\sigma=1000$](image)

As expected the lower the value of $\gamma$, the lower the processing time is for the same parameters $n$ and $m$. Consequently, looking at the performance alone, there is no limit for the number of enquiries that can be run following a single encryption of the identities in the system by the sender (Phase A of IDAP). But there are possibly security limitations to this procedure and the maximum number of subsequent
enquiries \( k \) that can be run out of a single encryption of a directory. These will be discussed later in this chapter (Section 6.3.1).

The graph shown in Figure 6-9, also illustrates that when condition from Eqn. 6-6 are met, the processing time is almost constant. This knowledge can be used to maximise the return-on-investment, thus fine-tuning the platform to use the maximum computational capacity of the protocol. Consequently, to provide maximum number of records with minimum effort:

\[
\gamma \times k \leq 0.1
\]  

\text{Eqn. 6-6}

Figure 6-8 Detailed comparison for IDAP and PE with.
For varying \( k, n=1\text{million}, \text{ and } m=10 \)

Figure 6-9 IDAP’s performance for different values of \( \gamma \), as compared to PE.
For varied \( \alpha \), \( n = 1 \text{ million}, \text{ and } m = 10 \)
6.2.4 Evaluation of IDAP with directory of identities

In theory it should be possible to improve IDAP’s performance in scenarios where directory of identities in the dataset is available. In such cases there is no need to encrypt the whole dataset with a commutative encryption scheme. Figure 6-2 shows that commutative encryption is the major factor in the total processing time, thus reducing the number of records that need to be encrypted is likely to lower the processing time.

The results of this experiment are shown in Figure 6-10 and Figure 6-11. As expected, with access to a directory of identities, IDAP performs significantly better for $\gamma < 1$ (Figure 6-10). This is for cases where the product of the number of interesting records, and dilution factor, is smaller than the number of records in the dataset. However, Figure 6-11 depicts that IDAP using a directory is only more efficient than ordinary IDAP for $k < 10$ under the test conditions ($n = 1$ million; $m = 100$; and $o=1000$). To be more precise, this occurs if condition from Eqn. 6-7 is met:

$$\gamma \times k < 1$$

Eqn. 6-7

Figure 6-10 Performance gain for IDAP run on dataset with a directory. Plotted for varying $m$, $n=1$ million, $o=1000$, and $k=1$
6.2.5 Use of dilution factor with different protocols

The technique of dilution the enquiries can also be applied to other SPIR protocols, as long as it operates in a single database scenario. As an example the combined approach formed from PEqT and OT protocol presented in Chapter 4 can be trivially modified to perform such enquiries, as presented in [9]. Figure 6-12 depicts a comparison of how IDAP compares to PE- and OT-based approaches described in Chapter 4, as well as OT-based solution that benefit from the notion of dilution. (Large preparation time of the OT-based approach is omitted here, and for this reason, the results are comparable to those of other SPIR protocols.) Looking at the $\gamma \leq 1$ range it can be seen that the introduction of the dilution factor to the PE protocol in IDAP cut the processing times nearly by half. However, the notion of dilution factor provides even larger benefits to non-commutative solutions, such as this based on the OT protocol.
6.2.6 Controlling the balance between privacy and feasibility

One of the key objectives of this thesis is to provide a level of control for the balance between privacy and feasibility. The discussion provided in Chapter 4 makes it apparent that even the modern SPIR protocols are unable to retrieve records from large datasets, as their performance is highly dependent of the size of the dataset. For this reason, the notion of the dilution factor has been proposed in this thesis. However, while adding a dilution factor to a protocol such as the one based on simple 1-n OT presented in Section 4.5 has a significant effect on performance as shown in Figure 6-12, its effects on the PE protocol are limited. More precisely, the dilution factor used in PE can half the run times for a single enquiry. However, it is clear from Figure 6-6 that the dilution factor has got a direct effect on the performance. The higher the dilution factor, the lower the performance, while at the same time we know that higher values of the dilution factor $o$ carry greater protection of privacy. Therefore, it is possible to control the balance between the privacy and feasibility using the dilution factor. This becomes even more apparent in Figure 6-13 depicting the run times of IDAP with directory against the size of the dilution factor.
Figure 6-13 Processing time against the dilution factor \( a \), for IDAP with directory of records. \( (n = 1,000,000; k = 1) \)

6.3 Presentation of qualitative evaluation

The previous section evaluated the performance gain in IDAP as compared to other PET protocols, such as PE. However, this would be irrelevant if IDAP would not enhance the investigative data acquisition process. In order to establish the benefits that IDAP brings into this process, a survey among of privacy and security professionals has been conducted, and security evaluation is carried out.

6.3.1 Correctness and Security

IDAP is a modification of the PE protocols that has its correctness and security proofs provided in [63]. The goals and logic of IDAP and PE are similar, however, IDAP is streamlined to provide better performance than PE in the specific use scenario of investigative data acquisition. In order to evaluate the correctness and security of IDAP this goals, in the form of definition of inputs and outputs, need to be clearly stated [90]:

*Chooser's input:* set \( V_C \) containing IDs of interesting records.

*Sender's input:* set \( V_S \) containing IDs of the records in the dataset, together with extra information about these records – \( \text{ext}(v) \).
Output: \( \text{chooser} \) learns \( |V_S| \) (the size of the set \( V_S \)), \( V_C \cap V_S \), and \( \text{ext}(v) \) for \( v \in V_C \cap V_S \), while \( \text{sender} \) learns \( |V_C| \). \( \text{Proxy} \) learns only the sizes of the sets.

Normally both parties learn the sizes \( |V_S| \) and \( |V_C| \), as by default all the encrypted identities in \( V_S \) are send to the \( \text{chooser} \), while the \( \text{chooser} \) in order to find the interesting records among these encrypted identities and in order to decrypt the \( \text{ext}(v) \) for these records provides the \( \text{sender} \) with encrypted elements of the set \( V_C \).

There is no requirement by the public authorities to know the size of the dataset, but since there is now a way to run IDAP and avoid providing the authorities with the dataset size, this needs to be accepted as an outcome of the protocol. The fact that the \( \text{sender} \) learns the number of interesting records is beneficial in the data acquisition scenario, as the \( \text{sender} \) can then verify that the \( \text{chooser} \) follows the data acquisition notice that would previously outline the IDs of the interesting records, and under IDAP would specify the number of the interesting records.

IDAP is based on \textit{Shamir's commutative protocols}, a variant of PH protocol where the prime \( p \) is public and common between the communicating parties. An adversary with the knowledge of the ciphertext \( C \) and the prime \( p \) would need to solve the following hard problem to break the commutative PH protocol [44]:

\[
\text{Eqn. 6-8}
\]

\[
e = \log_p C \mod p
\]

Just like RSA, the ciphertext created using the PH protocol may leak some information about the input plaintext message. Therefore, this algorithm is suitable for uses where the input is formed from random data. This is the case in the PE and IDAP, as the commutative PH is used to encrypt hashed IDs of the records. While it is normally recommended to use padding schemes in any implementation of RSA[124], and thus PH implementation as well, the PE and IDAP mitigate this requirement by using fixed size hashes as the input.

The proofs of the correctness and security of PE can be found in [63], while IDAP has modified this protocol by introducing the following improvements:
- Lowering processing time, by narrowing the scope of the enquiry.

- Allowing for multiple selection criteria.

- Restoring the balance between the privacy of the innocent and the suspects.

In order to narrow down the scope of the enquiry IDAP splits the PE protocol into three parts. However, the only way the operations of the protocol are affected is the fact that under IDAP the chooser request extra information for only \((m \times o)\) records, rather than for the whole dataset \(n\). The main consequence of this approach in respect of security of the protocol is that the sender knows that there are \(m\) interesting suspects in the set of identities the size of \((m \times o)\). This could become an issue if the same request is run against a number of parties and the parties collude, but this thesis has shown that the investigative data acquisition process can be treated as a single database scenario, if requests are made against CSPs. So colluding is not possible.

On the other hand, for small organisations with less than 100,000 IDs, there is no need to narrow down the results. Consequently, in IDAP, the privacy of the suspect is affected by the dilution factor \(o\), and the sender's probability of guessing the interesting records IDs is \(1:o\) and not \(1:n\). As long as \(o\) is reasonably large, and the sender has no other sources of information about the suspects, the privacy of the suspects should not be unaffected.

IDAP allows for the multiple selection criteria by hashing together different selection criteria and using it within the PE protocol as an ID of a record. This does not affect the security of the PE protocol. On the other hand adding the semi-trusted third party – the proxy – in order to restore the balance between the privacy of the innocents and the suspects somewhat modifies the security of the protocol. The proxy filters out the records not classified as interesting from the sender's response. Assuming that the semi-trusted party behaves as expected, the security of the \(ext(v)\), the data records contained in the sender's database is information theoretic from the chooser's perspective. On the other hand, if the proxy and the sender cooperate, they can easily work out the identities of the interesting records. The main aim of IDAP is to hide those identities from the sender, however, currently, the identities of the suspects are provided in every data acquisition notice. Consequently, if the semi-trusted party cooperates with the sender, the only result that would reveal the information is
currently openly communicated to the dataholders. Still, it is important that the semi-trusted party is chosen so it does have an incentive in maintaining the privacy of the investigations and the data subjects.

6.3.2 Survey Results

A survey has been carried out for the purpose of qualitatively evaluating IDAP. Due to the nature of the subject, the survey was aimed at specific security and privacy professionals, as well as the law enforcement professionals. This means that the responses are from the practitioners that would likely be the individuals involved during possible roll-out of IDAP. The graphs illustrating the results can be found in Appendix D.

According to the results, the participants are aware of the encryption technologies used for storing data and for communications, but are sceptical about the positive impact the that introduction of digital technologies could bring to security or privacy during the investigative process. Some respondents suggested that while security techniques can increase the security and privacy in an investigative process, the availability of data can balance-out these benefits, as it will open new avenues to abuse the access.

Most respondents agree that currently a data acquisition request can breach suspect’s human and natural rights, by affecting the relationship between the suspect and the *sender*. Also, similar views were shared in respect to the effect that a data acquisition request may have on an investigation. On the other hand, the plans to provide the public authorities with direct access to CSP data were met with a mix of responses. Respondents agree that these plans can allow the maintaining of secrecy of investigations and can provide faster access times to urgently needed data. However, the respondents were unsure what effects the proposed changes will have on the privacy of the individuals-under-investigation. The more verbose responses show worries that the extended availability is likely to cause excessive use the communications data. On the other hand, most of the respondents agree that hiding the identity of the suspect from the provider of the data can protect integrity and security of investigations, as well as the rights of the suspect.
Most respondents would accept IDAP as a solution to the privacy concerns in investigative data acquisition, and would expect their organisations to accept it, as well. However, the more verbose responses suggest that there are worries about the correctness of the non-trivial implementation of any mathematically sound protocol. These worries are also reflected in the respondent’s attitudes towards introduction of the proxy, as the number of individuals that would accept this solution matches the number of individuals that are against it.

6.4 Conclusion

The experimental results obtained from the implementation of the PE protocol show the same trends as the results achieved via simulation of the PE protocol using computational complexity tables. Since, IDAP uses the same cryptographic mechanisms as PE, these results allow for the simulations to be used as the main way of evaluating IDAP’s performance in contrast to PE. Other research has used similar, but less precise, approach of evaluating the protocols based on the number of asymmetric operations required [85, 92, 107], as this is the most costly operation in most OT and SPIR protocols. Thus, using simulations to compare these kinds of protocols is a generally accepted practice.

Results achieved from the simulations of IDAP under different conditions show that the performance of the protocol is highly dependent on the three key parameters:

- $m$ – number of interesting records.
- $o$ – dilution factor.
- $n$ – size of the dataset.

To be more precise the performance depends on the ratio of $(m \times o)$ to $n$. This proportion is defined as $\gamma$ in this thesis. It is shown that the operation of IDAP is optimal for $\gamma = 0.1$, as, at this point, the processing capacity of the protocol is used to maximum effect without affecting the total time required to process a query. If $\gamma > 1$ then PE should be used for the operation, rather than IDAP, as under such conditions the IDAP protocol would call for more than $n$ records to be encrypted on
the sender’s end of the process. However, such scenarios should generally be avoided, as they cannot guarantee $k$-anonymity of the interesting records.

According to the simulation results, IDAP is uniquely placed for multiple enquiries that run against the same set of identities. This includes enquiries where IP addresses or telephone numbers are used to match the records. The number of enquiries run against a single set of encrypted IDs is referred to $k$ in this thesis. For low values of $k$ and $\gamma$, this is for $k \times \gamma < 0.1$, the processing time is constant. As expected, the lower the $\gamma$, the higher the benefits of using IDAP over PE. Additionally, for $k \times \gamma < 1$ IDAP has a smaller processing time than a single enquiry using PE. Thus, as long as $\gamma$ is small, a number of enquiries can be run using IDAP without encouraging any considerable costs, where under PE the computation costs would grow in-line with the number of enquiries.

The dilution factor can be used to control the balance between privacy and performance in the system. Generally, from the privacy point of view, this factor should be as large as possible, as it specifies how many records dilute the identity of each interesting record. However, the processing times are directly affected by the size of the dilution factor, and for higher values, the enquiries are often not feasible. On the other hand, it is interesting to see that the dilution factor can be successfully applied to other types of single database SPIR protocols. In fact, the use of the dilution factor benefits non-commutative protocols more, and can enable other SPIR protocols to perform on par with IDAP and PE in $m$-to-$n$ enquiries.

The survey conducted among of information privacy and security practitioners suggests that hiding the identities of the suspects during the investigative data acquisition process could enhance the privacy of the investigations and protect the human rights of the suspects. Also, most respondents would accept IDAP as a solution to the privacy concerns related to the acquisition process.
Chapter 7

Conclusions and Future Work

7.1 Introduction

This chapter concludes the thesis. It summarises the steps taken to achieve the objectives of this thesis, including the brief digests of the findings of the Literature Review, initial experiments conducted to further explore the field of SPIR protocols and their potential use in investigative scenarios, as well as the design, implementation and evaluation of the novel IDAP. This final chapter also provides solutions to the motivating scenarios defined in Chapter 1.

This thesis has made a number of contributions to knowledge. It defines a platform that can assist making the investigative data acquisition process more ethical. In order to achieve this it introduces SPIR protocols into this process, and shows that thanks to employing the notion of $k$-anonymity in such protocols it is possible to achieve a satisfactory level of performance when conducting requests from large databases. This approach is analogical to the technique of *dilution* used, on occasion,
by investigators in order to hide object of their enquiry by combining a number of requests for data together. Thus, this thesis introduces a *dilution factor* used to express the level of privacy and secrecy that a given investigation requires by specifying how many records are used to hide a single interesting record. Furthermore, this parameter can be used to control the balance between performance and privacy in an enquiry. In order to ensure fairness in the protocol and to reassure the public that the records used to dilute the enquiry cannot be decrypted by the authorities this thesis proposes the use of an independent party to monitor the enquiries, and filter-out the responses from the dataholders. The thesis also proposes a technique for making complex queries on the datasets with minimal impact on computational complexity. These aspects of the thesis will be now summarised.

Finally, this chapter discussed possible areas of the further work that can further develop and evaluate the concepts of SPIR-assisted investigative enquiries, as well as use the findings of this thesis in different areas of knowledge.

### 7.2 Achievement of objectives

This thesis set out to meet five objectives defined in Chapter 1. These have been achieved and discussed at various points of this manuscript.

**Construct a literature review within the PET sphere**

The literature review of the PET sphere is provided is Chapter 3. This review focuses on the measures that individuals can use to protect their privacy in information systems, as well as those that organisations can, and possibly should, use to improve the privacy of the data-subjects in operations on personally identifiable data. PETs are often misperceived as computationally expensive methods for obfuscating sensitive data. However, there are various technologies that can be classified as PETs [91]. For example, security measures such as access-control are also vital for protecting privacy of data subjects, and therefore the distinction between ordinary security measures and PETs is disputable. The literature review has identified that it is possible to use a PET protocols referred to as SPIR protocols to retrieve data from almost any relational database, in a private manner [86, 108-110].
A drawback to the use of SPIR protocols is that most require an index, such as the index of a row in a database table, of the interesting record as an input. In many cases a directory of the records in a dataset exist, for example in an on-line purchase of digital goods scenario, the seller can publish a list of products together with their descriptions, and the buyer can use the index of the item of interest to purchase it and retrieve it using SPIR [85, 92]. However, for the datasets that do not have a public directory often a separate PEqT protocol needs to be run in order to find out the index of the interesting record [63, 117]. Another drawback of most SPIR systems is that they are often designed to perform retrieval of a single record from a dataset of \( n \) records. So, in the cases where more than one record would need to be retrieved from the same dataset, the run-time of the operation would be linear to the number of interesting records. One of the protocols that mitigate both of these drawbacks is the PE protocol based on commutative cryptography that was originally designed to share extra information about common records between two or more databases [63].

It is worth noting that SPIR protocols are generally based of trapdoor function, thus, they are often characterised by a high computational complexity. Since, under SPIR, a great deal of records need to be fetched from the sender’s database in order to stop this party from identifying the records of the interest, the protocols often has high communicational complexity. The balance between the communicational and computational complexity, though, can be altered by the use of balancers [102, 104], as well as with the advances in the area of TC and the use of SCOP devices [78].

**Define set of requirements that data acquisition process must meet.**

In order to provide a PET-based solution to the privacy concerns related to the data acquisition process, it was necessary to define the requirements that such a solution would have to meet. The process of selecting and refining these requirements, mainly based on available literature, is described in Chapter 4:

**Req. 1**  Allow for the gathering of multiple suspect records per enquiry, or have low overhead per each additional query run on the database.

**Req. 2**  Keep the data controller in charge of the data. A data record cannot be transferred or made available, to the public authorities, without the data controller’s verification of the request.

**Req. 3**  Allow for efficient and timely data retrieval. (The protocol run excluding
preparation should take less time than 30 minutes that it currently takes investigators to obtain investigative data in emergencies.

**Req. 4** Be cost-effective, as the platform will need to be deployed by a variety of organisations.

**Req. 5** Retain an audit trail of the processing performed on the potential evidence.

**Req. 6** Gain acceptance from the general public.

**Req. 7** Handle large datasets (such as datasets with more than 15 million records).

**Req. 8** Provide a technique for multiple selection criteria of interesting records, and allow for fuzzy matching on the selection criteria different than record ID.

*Construct a novel methodology for privacy-preserving investigative data acquisition.*

In Chapter 5 a platform for gathering investigative data from third parties has been proposed. The platform is called IDAP (Investigative Data Acquisition Platform) and it is well matched to meet requirements listed above. The design of IDAP is influenced by the discussions and simulations presented in Chapter 4. Based on the gathered requirements, SCOP-based solutions were considered to be more expensive and harder to deploy than software based equivalents. The literature review identified that most SPIR protocols can be combined with a PEqT protocol to form a system capable of searching the datasets belonging to third parties and retrieving records in a private manner. Such a combined approach was tested against the PE protocol that natively provides this functionality. Results of the simulations described in Chapter 4 have shown that it is unlikely for any 1-n SPIR protocol to perform better than the PE protocol in scenarios with multiple interesting records.

PE has become the basis for IDAP, the novel methodology for privacy-preserving investigative data acquisition. In order to address all the requirements for the data acquisition platform, IDAP modifies the way that PE handles requests by adding the following, domain specific, improvements:

- Introduction of the dilution factor that can limit the scope of a request in order to improve performance.
- Introduction of a method for performing low-overhead dataset searches with multiple selection criteria.
• Innovative use of a semi-trusted third party in order to restore the balance between the personal privacy and the secrecy of investigations.

These allow IDAP to become a scalable platform for privacy-preserving retrieval of investigative data from third parties. If IDAP is treated as a tool exchanging data between the relevant SPoCs, it fits well within already established, and well defined, processes for data acquisition, such as the code of practice presented in [8].

Propose an evaluation framework suitable to assess performance of novel cryptographic enhancements to retrieval of investigatory data.

The literature review has found that most researchers use the notion of complexity to evaluate OT- and SPIR-based privacy-preserving protocols [87, 88, 90, 92, 96, 108, 133, 137]. However, often only the number of computationally expensive encryption operations is included in the function describing complexity of a given protocol. In this thesis, a more precise approach is proposed. IDAP is evaluated based on the simulations plotted according to the complexity tables that depict cryptographic operations used per step of the protocol. With this approach it is possible to compile graphs showing IDAPs performance under various conditions.

Investigate parameters that could be used to assess the balance between the privacy and feasibility.

PET technologies are often computationally expensive, and the higher levels of privacy are usually linked to the higher complexity of protocols. Typically, it is difficult to control the levels of privacy offered by SPIR protocols. Typically, 1-n and m-n SPIR protocols aim to provide total anonymity, as the sender is unable to distinguish the interesting records from any other records in the dataset. However, the performance of the protocol depends mainly on the size of the dataset, which, in case of data acquisition enquiries, can be large. This thesis has identified that most enquires require only a single data provider. For example, if the last known location of a suspect needs to be established based on the data from a mobile telephony provider, then only one provider needs to be queried for this data, as it is based on the telephone number, it is possible to identify the right provider of services. Consequently, it is possible to limit the number of records retrieved from the provider to the interesting records, plus a number of records to obfuscate the
identities of these interesting records. Thus, it is possible to provide a constant ratio of interesting records \( m \), to the number of records that is retrieved from the provider \( o \). In this thesis this is referred to as the dilution factor and it provides a form of \( k \)-anonymity in the investigative scenario. Finally, it is possible to control the balance between the privacy and feasibility using the dilution factor, which is one of the main contributions to knowledge of this thesis. Currently, the dataholders know with 1:1 probability the identity of the interesting records, with 1-\( n \) SPIR protocols this would be a 1:\( n \) probability and would depend solely on the size of the database, while the dilution factor allows for custom 1:0 hiding of the interesting records. Needless to say, the dilution factor can be dynamically set depending on the characteristics of the enquiry. If an enquiry is urgent, then a low dilution factor can be set in order to speed up the processing.

7.3 Motivating scenarios with solutions

Chapter 1 has provided two motivating scenarios that helped to illustrate the problems in the current data acquisition process. The solutions to these scenarios are provided below. For these, take into considerations that the processing times discussed below are taken from the complexity tables, and the trial runs of the encryption protocols used to build IDAP. It is possible, though, to improve on these if the methods for fast exponentiation are employed and the implementation is done in a compiled C programming language (as shown in [90]).

**Scenario 1 – Request for ISP subscriber data:**

When using IDAP to retrieve data from an ISP the provider would not be given a list of interesting IP addresses, which are the addresses of the potential suspects in an investigation. Thus, the rights of the individuals-under-investigation should be preserved. Since, on the other hand, the data acquisition notice would be served under RIPA, it would not have to carry any justification to the dataholder, and with no identities linked to the notice it would be unlikely to compromise an investigation. Performance-wise, the IP address assignment will tell investigators the name(s) of ISPs that provide these IP addresses. This also means that a directory of the sender's database, namely the list of IP addresses served by a given ISP is public, or can be inferred with certain probability (although the percentage of unused addresses for a
given ISP would have to be known). The scenario specifies that there are 14 interesting IP addresses. Let assume that the dilution factor can be as little as 1,000, since such enquiries are common and the data holder does not know the reason for the request. If all IPs are provided by the same ISP, then IDAP is run against $14 \times 1,000$ records. Such an enquiry would only take 10 minutes to complete under the given parameters. Thus, the records would be returned within the 30-minute time window that under currently is achievable only in life threatening situations, as discussed in Chapter 4.

**Scenario 2 – Banking transaction details:**
In this scenario, if there is yet no official warrant for the enquiry, the request would need to be made under DPA. In such a case, the data controller can use the voluntary disclosure mechanism of the Act to provide investigators with the relevant data. However, the investigators would need to inform the data controller about the nature of the investigation, in order to persuade them to disclose the data records. But, there is no technical reason for the data controller to know the exact identity of the interesting record. The request should, in fact, be considered based on the circumstances described by the request, and not the identity of the suspect. Thus, by making IDAP enquiry for the records related to a given credit card number, the relations between the data-subject and the bank would not be affected, however, the investigation could be compromised as its details need to be disclosed. Thus, if the secrecy of the investigation is important, an IDAP enquiry could be made under a court warrant, still hiding the identity of the interesting record, with the warrant in place the data controller could not question the enquiry.

As to performance, at the time of writing there were 58 million credit cards in UK [138]. The initial few digits of the credit card can easily be used to narrow down the bank of the potential suspect. With at least 10 different banks offering such cards in the UK it is more than likely that the initial digits of the credit card (that specify its type and the issuing bank) the search could be narrowed down to less than 10 million records. Therefore, it would take the bank approximately two and a half days to build an encrypted dictionary (using an ordinary computer similar in the specification to the one used to provide experimental results). If the directory of the card numbers is already encrypted then the inquiry itself would take 38 minutes for a dilution factor
of 100,000, or less than a minute if the dilution factor is 1,000 (note that with the use of streamlined implementation of IDAP the processing time would be at least a magnitude shorter).

### 7.4 Contribution to Knowledge

This study presents a methodology for retrieving investigative data in a private manner. The following contributions to knowledge were made by this thesis:

1) This thesis demonstrated the manner in which SPIR techniques can be used to assist public authorities in privacy-preserving retrieval of investigative data from third parties. It has been established that the current processes of the data acquisition attempts to minimise the collateral damage that can be caused by investigations. However, they are unable to hide the identity of the suspects, and thus stop short of protecting privacy. The novel approach to performing investigative data acquisition presented here can add this ability to the already tried and tested data acquisition framework. Therefore, it can be used as a tool that can provide an enhanced level of privacy during the acquisitions, without the need to redesign the already established and well-defined process.

2) The problem of investigative data acquisition has been reduced to a single-database SPIR problem. The simulations presented in Chapter 4 have shown that a SPIR-based system for the data acquisition is not feasible if all the records in a large database must be processed. However, it was found that, in most investigative scenarios, such as those presented in [139], the investigators would need to make a single acquisition request per given identifier. For example, if phone billing information is required for a given phone number, the investigators could identify a relevant dataholder from the number allocation that is publicly available for call routing purposes. Consequently, with certain exceptions and precautions, the data acquisition process can be treated as a single database SPIR, and therefore $k$-anonymised queries that use the dilution factor can reduce the complexity of the protocols without leaking any data about the suspects, or to be more precise hiding each suspect in a group of records.
3) Chapter 6 evaluated the performance of IDAP, while its legal aspects where discussed in Chapter 5. This thesis has shown that it is possible to deploy IDAP into the real-life scenarios in order to benefit the privacy of the data-subjects and the secrecy of the investigations.

4) Section 5.3.1 of this thesis introduces a novel approach of employing $k$-anonymity principles in SPIR protocols. This is done in order to improve the performance of single-database SPIR systems. The main benefits of using such an approach are presented in Section 6.2.5 that illustrates performance of IDAP against PE protocol, and that of $1-n$ OT-based solution against its modification incorporating the dilution factor.

5) Definition of a technique, for building complex privacy-preserving enquiries has been provided in Section 5.3.2. It is based on hashing multiple selection criteria together to form one value that can be compared using $m$-to-$n$ PEqT protocol. For this reason it carries almost no additional communicational or computational complexity, however, it may put a larger strain on databases.

6) The key concept in enhancing privacy is to also enhance the perception of the final system by the individuals it aims to protect. This thesis proposes that encrypted requests should be relied on by an independent semi-trusted third party (Section 5.3.3). This party would ensure that the investigators can only get the data specified as interesting at the start of the investigation, thus, it would thwart any potential cryptographic attacks by the investigators on the data-records unrelated to the investigation, as these would be removed from the communication stream between the dataholder and the investigators. It is hoped that this technique should gain support of the general public.

7.5 Critical analysis

This thesis has focused on showing that it is feasible for the public authorities to use privacy-preserving techniques while retrieving personal data from third parties. There are various motivations for the public authorities to request third-party data. It
may be required by ambulance crews to pinpoint the location of a casualty using data from a mobile communications provider, or to find out the last transactions on a credit card that is being used by a suspect. The thesis focuses on the investigative scenarios as they carry a greater, and more evident, risk of privacy and human rights violations. However, the resultant IDAP can, and should, be applied to most requests for personally identifiable data made under RIPA and the voluntary disclosure mechanism of DPA.

This thesis employs the notion of $k$-anonymity to form a dilution factor for investigative data acquisition enquiries. With this factor, the balance between the privacy and performance can be dynamically controlled by the requesting party. For example, the Designated Person that scrutinises the requests for data [8] can decide the level of privacy and secrecy that given request should involve by manipulating the dilution factor. Such approaches, using $k$-anonymity, are now becoming popular in the PET domain. The inspiration for the dilution factor was the use of the $k$-anonymity models in the statistical data mining [97]. However, it is important to consider advances in the use of such models. These include the application of $k$-anonymity to location based services, hiding historical, current or trajectory location of the mobile users [140]. In this domain it is apparent that the algorithms for choosing the right records to obfuscate the request are vital to the privacy considerations. In this thesis it is assumed that a random selection of $(a - 1)$ records that are distributed in the dataset uniformly with the interesting record is a sufficient approach. This should allow for making a single query about a record, however, if the query is for the same interesting record and was to be repeated, the dataholder could possibly infer the identity of the interesting record. A solution to this problem could be similar to the one used in the SCOP-based PIR described in [78]. Namely, every new request can include a number of the identifiers that have previously being requested. In this way a constant number of records would overlap between the requests, and the dataholder would be unable to infer any extra information from overlapping requests. However, this thesis does not provide enough detail about the algorithm for selecting the records that obfuscate the true object of the enquiry, and this is a potential for further work.
In Figure 6-13 it is clear that whether IDAP is feasible or not depends on the dilution factor $o$. The higher the value of this factor, the higher the run time for the protocol is, and the less practical the use of IDAP. However, it should be noted that Chapter 6 provides results from simulated runs of IDAP, based on the complexity tables and empirically-obtained measures of the time required to perform different cryptographic operations. However, it should also be noted that the time measurements for different cryptographic operation have been obtained from Microsoft .NET C# implementation of the encryption techniques. It is certain that with use of machine compiled programming language such as C or C++, the performance of IDAP would be at the least a magnitude better. This is confirmed by the results achieved by other researchers [90]. Additionally, under IDAP, both parties can perform some of the processing in parallel, and the cryptographic operations themselves can also be parallelised. Thus, the performance of IDAP will depend on the hardware used to implement it. Consequently, the level of privacy that the framework can offer will depend on the monetary investment into IDAP. This confirms that privacy is not free [89], however, the monetary cost of computational power is decreasing.

The advances of cloud computing allow companies and individuals to purchase computing resources on at ad-hoc and pay-per-hour usage [141]. Running of IDAP in a cloud is not recommended if the data is stored in-house, but if this data is already in the cloud there is no reason why the computational power of the cloud should not be harnessed to perform IDAP, where it is possible to purchase High-CPU On-Demand Instances of virtual machines for less than 50 pence per hour [141]. Thus, an operation that would usually take hours to complete can be completed in the space of minutes, if 60 instances of such virtual computing machines are used at a cost of £30 per hour. However, the availability of cheap computing resources is also a threat to the system, as the perpetrators can also harness these resources. This is already evident on an example of cloud-based services offering cracking of WPA keys under the cover of penetration testing services [142].

The cost of communications should also be considered when discussing IDAP. This cost depends on the dilution factor, just as the cost of processing does. Thus, for low values of $o$, such as 1,000, the cost of communications should be reasonable.
However, where higher degree of privacy and secrecy is required, the costs of on-line communications could prove to be prohibitive. In such cases, it would be possible to exchange encrypted data via the post or couriers, as there are a small number of communication rounds between the parties.

This thesis suggests that commutative encryption protocols can provide for efficient privacy-preserving \(m\)-to-\(n\) equality tests, and also privacy-preserving information retrieval. While 1-to-1 PEqT protocols based on homomorphic encryption, such as [110], perform as well as 1-to-1 equality protocols based on commutative encryption, in scenarios with multiple records commutative encryption performs significantly better (Section 3.3.5 provides more detail). The same applies to the SPIR or OT protocols, while 1-\(n\) protocols based on commutative encryption perform on par with other 1-\(n\) SPIR methods. In this, the commutative protocols have a large advantage in \(m\)-\(n\) protocols, as only a single encryption of the records in the dataset is required for multiple interesting records, while with virtually any other encryption technology these records would need to be encrypted \(m\) times. For this reason, IDAP is based on commutative encryption. However, even though this type of encryption has been around for a few decades, as it was first scrutinised by Shamir in [50], it has not been explored and evaluated to the level that would allow a commutative encryption algorithm to become an acceptable encryption standard. Thus, before IDAP could become recognised as a suitable solution to mitigating collateral damages in investigative data acquisition scenarios, the commutative protocols employed and the logic of the protocol would need to be scrutinised by cryptanalysts. So far IDAP has been peer reviewed in [9, 12, 13], but in the area of information security this should be treated as a basic \textit{sanity} check. It is likely that flows will be found, in the protocol or its implementation [70, 120]. For example, in [57], Weis points out that most commutative encryption schemes, including the one presented in this thesis, are not semantically secure. This is caused by the very nature of the commutative encryption expressed by Eqn. 2-3, and cannot be avoided if this property is required. It can be assumed that the reason the commutative encryption has not been evaluated in the fashion that other encryption protocols are, is the fact that under commutative encryption certain links of the ciphertext to the plaintext are desirable. However, the main contribution to the knowledge of this thesis, are the three improvements proposed to make PE meet the requirements for IDAP. These improvements can be
applied in most SPIR protocols and, for this reason, even if commutative cryptography is rejected by the information security community as a basis for IDAP, the findings of this thesis can still be used to build IDAP based on another SPIR protocol. This should be feasible as Figure 6-12 shows the use of the dilution factor allows an approach based on 1-n OT to perform just as well PE does for a certain range of $\gamma$.

The commutative encryption PEqT protocol employed in IDAP compares hashes of the identities from the set of suspects to the hashes of the identities in the sender’s dataset. Therefore, there is no room for error. The identities need to be an exact match, so matching on values such as telephone numbers and IP addresses is preferred. There is no easy way to make IDAP match a different spelling of a name, such as John Smith and John Smyth. However, during the investigation of the subject area, a computationally-expensive method for achieving fuzzy matches has been developed and presented in [13].

7.6 Main findings

The current processes for data acquisition are designed to minimise the probability of collateral damage to the suspects and the investigations [8]. Still, the identities of the data records need to be provided to the dataholders, simply to identify the interesting data records. IDAP fits within this current methodology as it can provide the means to request data without revealing the identities of the interesting records. Thus, it is a tool that SPoCs can use to communicate, that also allows an independent watchdog organisation to monitor the exchange of acquisition notices and data. Consequently, the internal processes associated with data acquisition do not need to change beyond the fact that, under IDAP, the dataholder provides a large number of records as an input to the protocol, with the records locked by the encryption process in a way that renders them unusable to anybody who does not know the relevant encryption keys. The requesting party can then unlock only the records defined as interesting in the privacy-preserving data acquisition notice. In order to make the process more future-proof, and to gain acceptance of IDAP in the society, the watchdog organisation filters out the diluting records from the response, itself knowing only the encrypted form of the identifiers for the interesting records.
The operations of IDAP must not affect the validity of the gathered information as potential evidence for the use in a court of Law [20]. Since IDAP uses a number of cryptographic techniques to retrieve and decrypt the interesting records, some lawyers could potentially question the validity of the evidence, and lower the status of the data gathered to second-hand, or hearsay, evidence [139]. One solution could be to get the dataholder to retain the original records provided under each enquiry for a set period of time. However, since the dataholder does not know the identities of the interesting records, then all the records provided as an input to IDAP would have to be stored. This would be excessive, and a costly exercise. For this reason, under IDAP, the proxy is responsible for retaining the records identified by the public authorities as interesting, while the dataholder retains only the commutative encryption key used to lock the data records (there is only one per enquiry). Then if a verification of the evidence is required, these two parties could be made responsible for working together to validate the data.

It is interesting that from specifying who and why needs given piece of data some knowledge may be inferred. For example if a law enforcement officer requests to know the last location of an individual it is likely that the individual is a suspect, while if the emergency services request this information most people would assume that the individual is a casualty. With IDAP using a watchdog organisation as a central hub for all enquiries (Figure 7-1), the requests from investigators will be diluted by requests from other public authorities, including the health authorities and this will contribute to the privacy levels provided by the system.

![Diagram of request originators and data flow]

**Figure 7-1 Hiding of request originators can improve privacy**
This thesis is a response to the perceived trend of limiting privacy of individuals in order to protect the security of a nation. An integral part of the problem of privacy is the public’s perception. Therefore, the privacy, just like beauty, is in the eye of the beholder, and it is vital for any privacy measures to be acceptable by the individuals they try to protect, otherwise they are meaningless. This is the reason why achieving privacy is a complex matter beyond any technological solution. For this reason, the following paraphrased statement about security could also be used in relation to privacy:

“If you think technology can solve your security problems, then you don't understand the problems and you don't understand the technology!”

Ferguson and Schneier, [143], pp. XXII

This thesis suggests (in Chapter 4, and then in Appendix A), that security risk management can be employed to predict and manage risks to the privacy. However, this can only be done to a certain extend, as privacy, just like security, is a people problem [144]. Because of the human involvement of the both sides of the parameter, there is no way to predict all the vulnerabilities and potential threats to individual’s privacy, and it is far harder to predict individual’s perception of the privacy measures applied. Therefore, IDAP has been qualitatively evaluated with the help of a number of carefully selected individuals, and the results show the use of IDAP to perform data acquisition will be welcomed.

It is crucial that the merits of IDAP are communicated well in order to get as high a level of acceptance as possible. Professor Burkhard Schafer, Professor of Computational Legal Theory at Edinburgh University, when asked to give feedback on IDAP suggested an analogy to the “I am Spartacus” defence [145], for the way this platform protects the privacy of individuals under investigation by making a large group of individuals appear as suspect. If the size of the group is large enough then no repercussions can be applied and therefore the third party is unable to act on the information provided. This kind of defence was recently used by Internet users around the world in a protest against the judgement on a case of a Twitter airport
bomb threat joke [146], showing that this type of defence is potentially acceptable by the society.

IDAP, as defined in this thesis, is a feasible approach to acquiring investigative data from third parties, and such privacy-preserving solutions are generally welcomed by the governing organisation, the ICO [134]. However, the platform’s computational and communicational complexity may limit its use in the most urgent scenarios. Therefore, some alternative arrangements may be necessary for enquiries made to organisations that are required to provide investigative information on a daily basis, to the level that justifies employing full-time personnel just to handle requests. In such a case, strong privacy-preserving procedures may be put in place, with certification system for organisations and vetting programme for the individuals handling the requests for investigative data.

### 7.7 Future Work

There are a number of important areas of further investigation, and different applications for the protocols discussed. The areas of future work include:

- Evaluate database load. Searching databases on columns containing record IDs is usually a fast operation, since such columns are usually indexed by the database system. This thesis proposed a technique for privately matching records on multiple selection criteria, however, these criteria would potentially include columns that are not indexed. Use of this technique may cause considerable load on the database and therefore needs to be evaluated.

- Investigate of a technique for selecting an appropriate dilution factor and diluting records depending on the circumstances.

- Include warrant signing and verification system in IDAP. Similar system already exist as described in [86].

- Evaluate an add-on to office packages, such as Microsoft Office and Open Office that could perform IDAP on spreadsheets and thus it could enable small organisations to utilise IDAP.
Investigate using the components of IDAP in privacy-preserving proximity- and location-based dating and social networking. There is a high degree of risk to privacy of individuals actively looking to find a partner or expand the network of people they know. With some components of IDAP, it would be possible to actively advertise one's own preferences using a mobile device, allowing for the individuals matching the profile and advertising similar values and interests to recognise each other.
Chapter 8

References


[137] F. Olumofin and I. Goldberg, "Privacy-preserving Queries over Relational Databases," in *Privacy Enhancing Technologies Symposium*, vol. 6205,


Appendix A

Risk
Security is a people problem, not just a technology problem.

(Wadlow, [144])

Most technology issues have absolute solutions. It is possible to design a production line robot to perform certain task 24/7 with no downtime, nor errors. Such robot could be tested in every scenario that could possibly arise. However, it is impossible to provide absolute security, mainly because there is no way to predict all the possible avenues of the attack on the system. This is because of the human involvement at both sides of the security parameter [144]. Surveys discussed in [147] expand on this point by showing that management controls are the poorest implemented in contrast to technical and operational controls. Consequently, human add random unpredictable results to any system and providing absolute security is impossible. The previous section has introduced privacy as being related, or sometimes considered to be highest level of, personal data security. This section provides description of the decision making process required to achieve data security.

Security and privacy principals need to be implemented into information systems and understood by the system users. The way they are enforced depends upon the actual requirements for a particular system, the system itself and, just as anything else in the material world, on cost-benefit ratio of the implementation. The choices are made by first quantifying risk, i.e. risk analysis, and then exploring possible controls/countermeasures against these risks, i.e. risk management. Risk in the information security field is the possibility of loss, damage, or any other undesirable event affecting assets. The risk analysis process involves identifying assets within an organisation, recognising vulnerabilities that could potentially expose these assets and threats that can exploit the vulnerabilities to impinge upon the assets [148]. The assets could be material, such as server estate, or more abstract, such as good name of the business, thus assets that indirectly affect the wealth of the organisation. Also the vulnerabilities and threats could have more abstract than material forms.
Consequently, risk to privacy of the data subjects can be quantified by following the risk analysis process, while the risk management process could help introduce appropriate cost-effective countermeasures against possible threats.

Risk analysis and management processes are jointly referred to as risk assessment. In order to define an appropriate security strategy for any system dealing with sensitive information, it is vital to perform risk assessment so that they can better manage the risks. Managing the risks can be performed by a careful system design and post implementation controls, such as security policy deployment and monitoring [43, 149-151]. There are two goals in the context of this research that can be achieved by performing above tasks:

- Produce guidelines for system design
- Identify the events and the elements of the system that need to be monitored
- Define events which should not be kept in the system audit trail

Whilst the two goals are clear, the last needs additional explanation. First of all, there are concerns over the audit of information systems that were already mentioned in the previous sections, hence various legislations forbid monitoring of certain information flows [1, 17], also cases of the national security where the investigators request that the inquiry should be kept off he record should not be included in the audit trails [10]. Auditing systems can produce vast amounts of information, which can lower the efficiency of process being monitored if implemented inline with the core operations or can reduce the court admissibility of the monitoring logs if the monitoring is no in line and audit trail is incomplete due to overloading of the monitoring system [19, 152]. For this reason, it is advisable to consider excluding some data operations from the audit trails.

**A.1 Model based risk analysis**

Craft et al. [153] defines six elements of risk that needs to be addressed whilst performing a risk assessment: assets; vulnerabilities; threats; impact; likelihood; and safeguards. However, whilst discussing these terms one must remember to consider factors such as: technology; organisation; environment; and, of course, people [154].
Thus, these elements are further expanded by Ozier [155], in order to include a more detailed breakdown. Consequently, unmanaged risks may be determined as a product of threats, vulnerabilities and the value of assets [156, 157] as illustrated in Figure A-1 (a).

There are many ways to perform risk analysis, and the selection of the right one depends on the organisation, and the information system it uses [150]. There are quantitative and qualitative methodologies to this problem, both with their advantages and disadvantages. The quantitative approach requires the use of complex mathematical equations and its effectiveness depends on the precision of defining a factor of probability [158, 159]. Since this probability factor is often loosely chosen, the results of quantitative risk analysis may be deceptive, and thus, qualitative methodologies are considered to be more adequate for analysing complex systems.

Qualitative methods, although easier to implement, require a team of specialist, from different areas of organisation to produce a collective strategy. Organisation of such a team, and the accumulation of individual expertises may become almost impossible to manage, especially in situations where every team member wants to prove the significance of their department. Thus, the results of qualitative analysis often depend on the experience of the assessment team members, and the data available [156]. For these reasons many tools have been designed to assist in the process by structuring the team meetings, and by providing data-primed, pre-programmed, case scenarios. Currently, there are three major risk analysis methods available:
• CAMM – designed by the Central Computer and Telecommunication Agency (now Office of Government Commerce), in 1985, to provide a methodology for reviews of information systems security in UK government departments. Since then it has been redeveloped by Insight Consulting to become a total information security toolkit [150, 156].

• OCTAVE – a risk-based, strategic assessment and planning technique for security developed by CERT, a division of Software Engineering Institute at Carnegie Mellon University [150, 160, 161].

• CORAS – a method for Model-Based Security Risk Analysis (MBRA) developed by an EU-funded international team of specialists. The work on the project started in 2001 and was targeted at the application of MBRA in security-critical systems [149, 150].

While CAMM and OCTAVE use text- and table-based documentation, CORAS uses an easy-to-comprehend formal modelling language to assist the team who is performing risk analysis. For that reason CORAS has been chosen to illustrate decision making process describing selection of appropriate risk management strategy for the IDAP system. This is provided in Chapter 5.

A.2 CORAS risk modelling

The relationship between aspects of security risk analysis is illustrated in Figure A-2. The figure illustrates that selection the Target of Evaluation (TOE) greatly influences the risk analysis process. TOE describes assets of certain value that need to be protected from threats.
In project management risk is an umbrella term, with two meanings. Risk may be negative, thus a threat to an asset that is capable of exploiting vulnerability, or a threat to an objective of a project. However, in project management risk can also stand for an opportunity to improve on those objectives [162]. CORAS threats risk in the negative way, since CORAS was specifically designed for security risk analysis, which is somewhat different than ordinary project risk analysis.

The most important contribution that CORAS made to the field of risk management is the definition easy-to-understand UML language with symbols illustrating factors of risk. These symbols are shown in Figure A-3 and their usage is illustrated in Figure A-4. One could argue that these symbols are immature. However, these symbols are easy-to-decipher and –remember. For the members of western society these symbols are clear in the meaning, and even without the legend provided in Figure A-3, the threat diagram shown in Figure A-4 may be deciphered.

---

**Figure A-2 CORAS ontology** (Gran et al., [150], pp. 1498)

**Figure A-3 Symbols from the CORAS risk modelling language** (Braber et al., [149], pp. 108)
There are few different approaches to modelling risk, however, CORAS thanks to the easy-to-understand symbols and the use of UML is likely to benefit teams of experts from various domains looking for a common language to communicate during risk assessment tasks.

A.3 Return-on-Investment (RoI), Return-on-Attack (RoA)

Traditionally, risk management focussed on achieving as large RoI as possible. RoI of a safeguard illustrates the worthiness of applying this safeguard [163] and can be calculated as shown in Eqn. A-1. According to this commonly used equation the higher the efficiency of the safeguard compared to its cost, the higher the RoI and suitability of a safeguard in a given scenario. However, in [151] Cremonini and Martini highlight that RoI provides only a partial characterisation of investments since it lacks consideration from perspective of potential perpetrators. The drawback of RoI is a result of gross simplification that the gain to the intruder is equal to the loss incurred by the organisation as a consequence of the attack. Cremonini and Martini suggest that another index called Return-on-Attack (RoA) should be introduced to express possible gain from attacks to the intruder. Eqn. A-2 shows how this index could be calculated.

\[
RoI = \frac{\text{Efficiency} \times \text{Cost of Security Incident}}{\text{Cost of Security Measure Implementation}} - 1 \quad \text{Eqn. A-1}
\]

\[
RoA = \frac{\text{Gain from Attack}}{\text{Attacker’s Costs on Unprotected Asset}} \times (1 - \text{Efficiency}) \quad \text{Eqn. A-2}
\]
RoA is, thus defined as the gain the intruder can expect from a successful attack over the cost of overcoming the evaluated security measure. Consequently, RoI provides a measure of the suitability of the safeguard against a malicious attacker. Also, lower RoA gives smaller probability of an attack in the future.

**A.4 Lifecycle of Risk Management**

Security policy is one of the most important outcomes of security and privacy risk analysis. It provides a set of high level rules that when adhered to by all the parties involved ensure that appropriate levels of security and privacy are maintained. However, the main problem with enforcing security policies is the fact that they are drafted for abstract infrastructures and scenarios, and thus they can never be implemented precisely [164]. Consequently any security should be drafted for a specific system in mind and any parts of it that can not be implemented should be reviewed during a new round of risk analysis. Also, maintaining security, and therefore privacy as well, of an information systems is an ongoing process. Thus, once the system is in place scheduled and event triggered audits should take place to refine it. In [165], researchers propose a lifecycle as a basis for maintaining a required level of protection after the initial deployment of the system. Such approach agrees with the British Standards [166], as well as other risk assessment standards, such as Australian standard AS/NZS 4360:1999 that has been adapted by CORAS [150].
Appendix B

Simplified Operation of IDAP
Full code produced during this thesis is available upon request.
Appendix C

Empirical Evaluation Results
### m = 1, 50<n<1mln

<table>
<thead>
<tr>
<th>m</th>
<th>n</th>
<th>OT</th>
<th>PE</th>
<th>PE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>50</td>
<td>39.724083</td>
<td>2.457119</td>
<td>2.435369</td>
</tr>
<tr>
<td>1</td>
<td>100</td>
<td>79.339033</td>
<td>4.723569</td>
<td>4.701819</td>
</tr>
<tr>
<td>1</td>
<td>500</td>
<td>396.258633</td>
<td>22.85517</td>
<td>22.83342</td>
</tr>
<tr>
<td>1</td>
<td>1000</td>
<td>792.408133</td>
<td>45.51967</td>
<td>45.49792</td>
</tr>
<tr>
<td>1</td>
<td>5000</td>
<td>3961.604133</td>
<td>226.8357</td>
<td>226.8139</td>
</tr>
<tr>
<td>1</td>
<td>10000</td>
<td>7923.099133</td>
<td>453.4807</td>
<td>453.4589</td>
</tr>
<tr>
<td>1</td>
<td>50000</td>
<td>39615.059133</td>
<td>2266.641</td>
<td>2266.619</td>
</tr>
<tr>
<td>1</td>
<td>100000</td>
<td>792299.1091</td>
<td>45329.19</td>
<td>45329.17</td>
</tr>
</tbody>
</table>

### 1<m<50000, n=1mln

<table>
<thead>
<tr>
<th>m</th>
<th>n</th>
<th>OT</th>
<th>PE</th>
<th>PE</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>100000</td>
<td>983619.4886</td>
<td>45329.87</td>
<td>45329.84</td>
</tr>
<tr>
<td>10</td>
<td>100000</td>
<td>1222769.963</td>
<td>45330.71</td>
<td>45330.69</td>
</tr>
<tr>
<td>50</td>
<td>100000</td>
<td>3135973.758</td>
<td>45337.47</td>
<td>45337.45</td>
</tr>
<tr>
<td>100</td>
<td>100000</td>
<td>5527478.502</td>
<td>45345.91</td>
<td>45345.89</td>
</tr>
<tr>
<td>500</td>
<td>100000</td>
<td>24659516.45</td>
<td>45413.46</td>
<td>45413.46</td>
</tr>
<tr>
<td>1000</td>
<td>100000</td>
<td>48574563.89</td>
<td>45497.94</td>
<td>45497.92</td>
</tr>
<tr>
<td>5000</td>
<td>100000</td>
<td>239894943.4</td>
<td>46173.62</td>
<td>46173.6</td>
</tr>
<tr>
<td>10000</td>
<td>100000</td>
<td>239045417.8</td>
<td>47018.21</td>
<td>47018.19</td>
</tr>
<tr>
<td>50000</td>
<td>100000</td>
<td>2392249213</td>
<td>53774.97</td>
<td>53774.95</td>
</tr>
</tbody>
</table>

Results from the simulations are not included with this thesis, as it is possible to replicate them based on the complexity tables.
Appendix D

Survey
Currently, when investigators need to acquire data about their suspect from a third party (dataholder), it is necessary to identify the suspect to the third party. For example, if a police investigator would like to find out the list of transactions performed during the previous night by Bob Smith from his bank, the investigator would reveal to the bank that Bob Smith is linked to an investigation.

Can such a data acquisition request: 
- Compromise an investigation?
- Breach suspect’s human rights / privacy?
- Affect the relationship between the suspect and the third party providing the data?
The government plans to introduce a centralised system allowing investigators access to data collected by Internet Service Providers (as well as some other Content Service Providers). In such a system investigators will be able to access required data fast and in a secure manner. Additionally the ISPs will not be able to monitor enquiries made by the police.

Can this solution:
- Protect integrity / secrecy of an investigation?
- Protect human rights of the suspect and the relations of the suspect with the dataholder?
- Protect privacy of the innocent Internet users?
- Improve investigations by providing required data to investigators in a speedy manner?

We have developed a system that can be used by the investigators to hide the identity of the suspects during the data acquisition process from ISP or any other third party. Can hiding of the suspect's identity:

Can this solution:
- Protect integrity of an investigation?
- Protect human rights of the suspect and the relations of the suspect with the dataholder?

There are some drawbacks introduced by our system. The time required for the enquiry depends on the number of records in the third party database. For example: Data acquisition from a database of a thousand employee records would take approx. half a minute, whereas notice served to an ISP with around million users can take approx. 500min (8 hours) to be processed by an ordinary computer.

Do the benefits of this system outweigh this drawback?

- Investigators can accept these turnaround times in most circumstances.
- The third party should have enough resources to facilitate such lengthy data processing request.
It is possible to lower the turnaround time for processing data from large databases by operating on the subset of the database. In this way the processing time can be minimised, however, it would be easier for the third party to identify the suspect. Still the probability of third party finding out the identity of the suspect would be 1 in 1000 or 1 in 10000.

Do you think that this solution is acceptable from the perspective of:

- Integrity/Secrecy of an Investigation?
- Protection of suspect’s human rights and the relation of the suspect with the data holder?

In order for the privacy to be maintained the system encrypts a large chunk of the third party database with techniques similar to those used to transfer data securely over the internet and transfers it to the investigators. From these the investigators are then able to extract only the data of the suspect specified on encrypted acquisition request and nobody else. It can be proven that it is not possible for the investigators to extract any other information from the third party database. Do you think that the benefits of the added privacy and security, as well as secrecy of the investigation outweigh this drawback?

Some critics worry that the government agencies are capable of breaking encryption. In order to guarantee to the public that the investigators will only have access to the data requested we purpose to introduce another party into the system. An independent body, such as Information Commissioner’s Office, could filter the encrypted responses from the data holder, passing on only the data requested by the investigators. This party would not be aware of the investigation context, ID of the suspect, or the context of the response from the data holder. And as this party would have no intentions in finding these information, nor have the processing power to do so, it could be trusted not to attempt to break the encryption.

- You?
- Your organisation?
Appendix E

Publications
The following poster has been presented to the Scottish Institute for Policing Research institute (SIPR) during the annual conference, 2008.

**Privacy-Preserving Investigations**

*Procedures for gathering investigative data may breach human rights*

- Public Authorities may serve notice to a Data Holder (such as Internet Service Provider) to request information about a suspect. Such a notice would state the suspect’s identity, which could affect natural rights and privacy of this individual.

The procedures for data gathering could be enhanced, since security and privacy can be achieved together.

**Oblivious Transfer Access for Privacy-Preserving Investigations (OTAPPI) may be a valid solution**

- Data Holder retains control over the data access.
- Data transfers are secure.
- Processing time required is negligible in comparison to the length of an enquiry.
- Faster than traditional exchange of investigative data.
- Suspect’s identity is kept secret from the Data Holder.
- Privacy of the records in the Data Storage is maintained.
- Human rights of the individual under investigation are preserved.

**OTAPPI uses established security and privacy technologies**

- Open, platform independent architecture.
- A third trusted party is not required.
- Common encryption protocols are used to perform a symmetric private information retrieval.
- RSA is used to compare identities without revealing them.

Further investigation is required

- Examination of different Oblivious Transfer protocols.
- Evaluation using real data.
- Research of the legal issues.
- Survey among investigators.
Validation of 1-N OT Algorithms in Privacy-Preserving Investigations
Zbigniew Kwecka, Prof William J. Buchanan and Duncan Spiers
School of Computing, Napier University, Edinburgh, UK
z.kwecka@napier.ac.uk
w.buchanan@napier.ac.uk
d.spiers@napier.ac.uk

Abstract:
Most organisations rely on digital information systems (ISs) in day-to-day operations, and often sensitive data about employees and customers are stored in such systems. This, effectively, makes ISs enhanced surveillance measures, which can reach further than CCTV monitoring and provide valuable resources for internal and external investigations. For privacy reasons, if a digital forensic investigation is to take place, only the investigators should know the identities of the suspects. Ideally, the investigators should not have to disclose these identities to the data holders, while the data holders, i.e. organisations whose data subjects are being investigated, should not have to disclose their full databases to investigators. The only data that should be disclosed should relate to that involving the subject – thus the need for a privacy-preserving investigation system. Several privacy-preserving algorithms have been proposed, but most of them are only of theoretical interest since empirical evaluations have rarely been undertaken. The main novelty in this paper is that it applies a 1-out-n Oblivious Transfer (1-n OT) algorithm to a new area of privacy-preserving investigations. Hence, an implementation of a straightforward privacy-preserving investigation system that can be used in real-life applications is outlined. The system uses tried-and-tested encryption algorithms: RSA for hiding the identity of the suspect; AES to conceal from investigators records not relating to the suspect; and commutative RSA to allow discovery of index where a suspect’s data is stored in the third party records. This paper outlines an initial evaluation of the system proving that it may be successfully used in digital forensic investigations, conducted by public authorities and private organisations alike. The empirical evaluation also shows that the time required by this system to run grows in line with increasing number of records and increasing size of records, which is desirable compared to exponential growth observed in many systems that employ 1-n OT protocols.

Keywords: privacy preservation, data mining, digital forensics, digital suspect watchlist, Oblivious Transfer

1. Introduction
Most organisations rely on some form of digital ISs. These are usually used to record information about customers, employees and their interaction. However, such systems are often capable of, and used for, recording actions performed by the users, which makes these perfect surveillance measures. Thus, an organisation may record activities of system users, whether they are working in the office, or using software or equipment belonging to the organisation, while they are ‘on the go’ (Vlahos 2008). The majority of the population accepts monitoring as a required safeguard against threats to the organisations concerned and do not consider such method as a direct threat to their personal privacy (Gill & Spriggs 2005). At the same time, many people worry about the privacy of their digital records and the unfair use of their data (Young et al. 2006). Thus, there are two issues that need to be addressed to ensure that the privacy of law-abiding individuals is not affected by the surveillance systems in operation. The first issue relates to the fact that data obtained for legitimate reasons is often used for other purposes which could be deemed as intensifying surveillance and invasion of privacy (Ball et al. 2006). Arguably, the best course of action is to amend legislation accordingly. The second issue is linked to privacy concerns in access to the data collected, and this paper proposes a protocol to address it. In other words, it relates to internal and external investigators who request information on a third party’s data subject, in such a way that traditional investigations would typically invade privacy of the data subject (Frikken & Atallah 2003). The following scenario should illustrate this clearly:

Government security services, following a lead, investigate several potential suspects, including law-abiding Bob. Investigators need to identify Bob as a suspect in order to legally require a third party, an on-line shop where Bob made some recent purchases, to release the data. Later, Bob is removed from the list of suspects based on the evidence gathered. However, a few days later, Bob wants to use his favourite on-line shop again, but this time requests 0% finance on the purchase. His application is refused. What Bob does not know is that he was a suspect, and the on-
line shop has placed him on the list of high-risk borrowers, because of the recent enquiry from the security services. Most importantly, Bob is unable to find out why his application was refused, since disclosure of matters affecting national security and crime prevention are exempt from many provisions of the UK’s Data Protection Act 1998 (DPA) (sections 28 and 29). Given also that the DPA was enacted in implement of the European Data Protection Directive 95/46/EC, similar exemptions apply across the European Economics Area.

This scenario demonstrates an invasion of Bob’s human and natural rights, and, in this case, the party that caused the violation are the security services, as their actions made a third-party aware that he is a suspect in an investigation. According to the DPA, organisations may provide other organisations with personal and sensitive personal information about a data subject in some exceptional circumstances (see Part IV of the DPA). Additionally, in the UK, The Regulation of Investigatory Powers Act 2000 (RIPA) and The Regulation of Investigatory Powers (Scotland) Act 2002 (RIPSA) gave even wider regulated powers to investigators. For instance, emergency services may request information on the allergies of a casualty, and of a casualty’s relatives, from any institution that they suspect may have this data, and such institutions may lawfully disclose the data. Accordingly, the police and other public authorities may also request data related to their suspects, based on the same reasoning. Thus, in the above scenario, the security services and the data controlling organisation would act lawfully in accordance with the above legislation. However, their actions could seriously impinge upon Bob’s natural rights and quite possibly his privacy. In the scenario it has a very particular detrimental impact upon his rights concerning his future economic relations with the data controlling organisation. This raises interesting issues about the legal remedies, if any, open to him. In similar circumstances a case could be made out that there has been a breach of Article 8 of the Council of Europe’s Convention on Human Rights (now enforceable in the United Kingdom under the Human Rights Act 1998). This would be difficult to pursue for a number of reasons. Quite apart from the practical difficulty of knowing that there has been a breach of rights, how the breach has come about, who is responsible and how to prove it (what might be called “evidential difficulties”), there is also the question of the extent to which those responsible might be able to claim exemption from responsibility (which might be called “substantive difficulties”). The right of privacy under Article 8 (like most human rights) is a qualified right, meaning that a public authority is entitled to disregard the right where the interests, among others, of national security or the prevention of crime and disorder require. Such an exemption would normally exclude the possibility of Bob being able to pursue damages against the public authority. However, perhaps the correct approach is to regard the exemptions as only coming into effect where they are proportionate. If there is a way to obtain the evidence they require without invasion of privacy and other rights of the suspect and without the adverse impact the scenario predicts, then it is arguable that the public authority should take into account the rights of the suspect and so to choose the least disruptive method of obtaining the evidence they need for their purposes. It therefore could be argued that if they chose a method which invades protected rights and is likely to cause adverse impacts, then the public authority have used an exemption disproportionately and so should be obliged to recompense the suspect for the harm perpetrated by their choice of method. It is interesting to conjecture to what extent a court would entertain such a claim. Moreover, the security services may have compromised their investigation by revealing the identity of their suspect to the on-line shop and its employees.

The focus of this paper is on establishing how an appropriate level of privacy can be maintained in the processes of inter-organisational and inter-departmental personal data acquisition in such a way as to avoid both of the issues highlighted above. Thus, existing technologies that may assist public authorities and internal investigators in large organisations in gaining secure and confidential access to a suspect’s data are explored. The novelty here is applying scalable 1-n OT protocols into a complete investigative system and validating its use in real-life applications through empirical evaluation.

2. Existing Technology

In the UK there is little government guidance relating to the way data access requests should be sent by public authorities, or to the method of private data transfer back to the authority. RIPA specifies that such operations should be performed according to DPA (this is regulated by the Statutory Instrument “The Regulation of Investigatory Powers (Acquisition and Disclosure of Communications Data: Code of Practice) Order 2007” S.I 2197/2007 which came into force on 1st October 2007), which requires that data is secure, but it does not provide a specific method of achieving this. However, this
Largely based on the same functions as common encryption schemes, multiparty computation has a strong theoretical underpinning (Goldwasser 1997) and the structural building blocks for the creation of a solution to these privacy concerns (Frikken & Atallah 2003). Multiparty computation allows \( n \) different parties to engage in a protocol to enable them to compare their secret inputs, or to compute a function, without revealing these inputs. A classic case is Yao’s millionaires’ problem, where two millionaires seek to compare their fortunes without revealing the exact figures involved (Yao 1982). Yao provides three different solutions to the problem, giving the basis for the multiparty computation for \( n \) equal to 2. That is, when only two different parties are involved. He also provides a technique for scaling any 1-2 OT protocol into 1-n OT. Since Yao’s hypothesis, many other multiparty computation schemes for value comparison have been defined, including loosely proposed real-life alternatives gathered in Fagin, Naor & Winkler (Fagin, Naor & Winkler 1996). Goldreich, Micali & Wigderson (Goldreich, Micali & Wigderson 1987), later introduced schemes designed for general \( n \).

Two major types of protocols that may help to protect privacy, when retrieving information from large data sets, are Private Information Retrieval (PIR) and 1-n OT. In the PIR, a chooser, the party that requests information, may query a database in a way that the database provider (sender) cannot identify which row is being retrieved. A proposed solution to this problem is to provide the chooser with a copy of the database, since, in ordinary PIR, privacy of the records in the database is not considered a factor (Ostrovsky & Skeith III 2007). Consequently, the true aim of research in the field of PIR protocols is to achieve the lowest possible number of computations and communications during the private selection of a record. The use of ordinary PIR protocols would, therefore, solve the problem of compromising an investigation by a public authority. They would, however, negatively affect the privacy of the data subjects unrelated to the investigation, since their records would be provided to the authorities, along with the suspect records.

More advanced than PIR protocols are 1-n OT schemes, which are stronger than PIR on the basis that the chooser may learn only one row of the results provided by the sender. For this reason, 1-n OT is often called symmetric PIR. 1-n OT schemes, on their own, may be considered as having limited benefit (Schneier 1995). However, this paper argues that, when combined with other protocols, they may provide ideal tools for a privacy-preserving enquiry system.

Many oblivious transfer (OT) protocols were designed to allow the chooser access to a randomly selected record from \( n \) secrets kept by the sender. Such protocol is presented in Schneier 1995. This may be useful in mental games (Goldreich, Micali & Wigderson 1987) but it does not satisfy our criteria. Although some 1-n OT protocols allow the chooser to select the record of interest by means of an index, the chooser often does not know the relevant index. Returning to the scenario provided in Section 1, the security services would know the identity of the suspect, through a credit card number or an IP address, for example, but would not, however, know the index of Bob’s records in the database of the on-line shop. Consequently, they would be unable to request Bob’s records in a secure manner by using 1-n OT. The solution to this problem, once again, can be found in the field of multiparty computation. More precisely, multiparty computation offers a few alternative schemes for performing an asymmetric equality test (Frikken & Atallah 2003). This is a test that can be used to compare the secret inputs of two parties and provide the results to only one party. Therefore, before parties engage in 1-n OT protocol, a synchronisation phase may occur in which the chooser may discover the index of the suspect in the sender’s database. Using this technique, the security services from the scenario could compare Bob’s IP address to those of recent visitors to the on-line shop, in a way that none of the parties would know the IP addresses being compared. The security services would, however, know the indexes of the records being compared. Once asymmetric equality protocol has reported a match on the IP addresses being compared, the security services would know the index needed for 1-n OT enquiry. Frikken & Atallah describe a technique used for asymmetric equality testing that employ commutative encryption schemes. These schemes are characterised by the fact that a plain-text, encrypted with two different sets of keys into a cipher-text, may be decrypted in any order of applying the keys. Therefore, in a commutative encryption, where \( E_A \) and \( E_B \) are two commutative functions, the following is true:

\[
E_A(E_B(plaintext)) = E_B(E_A(plaintext))
\]
Equation 1: Commutative functions

This paper proposes a system that uses the characteristic shown in Equation 1 during the ID synchronisation phase, as described in greater detail in Section 4.2.

3. Related Research

In recent years, privacy-preserving data collection and mining have received a considerable amount of attention. Researchers have developed techniques that permit data mining within a cross-section of records rather than from the whole set (Aggarwal & Yu 2008; Agrawal & Srikant 2000; Kantarcioglu & Vaidya 2002). Some of these approaches are based on perturbations, as the one described in Agrawal & Srikant 2000, and can result in an information loss. This information loss is perceived in Aggarwal & Yu 2008 as a natural trade-off between accuracy and privacy, since the larger the number of perturbations, the greater the level of privacy. Another approach includes $k$-anonymity models first proposed in Samarati 2001, where attempts to link any gathered records to its owner creates at least $k$ different entries. Also, Aggarwal & Yu 2008 provide techniques based on transforming original data sets into anonymised data sets. Overall, these well-developed solutions are suitable for data mining in relation only to statistical information rather than to individual records, where the identity of the data subject is essential – applying them in a Court of Law would raise questions of authenticity, accuracy and validity, which the above well-developed privacy-preserving solutions cannot provide.

Current research into the retrieval of individual records mainly focuses on PIR or 1-n OT protocols (Frikken & Atallah 2003; Kantarcioglu & Clifton 2003; Li et al. 2008; Naor & Pinkas 2001; Wen-Guey 2002). Possible uses of these protocols include: electronic watch-lists of suspects (Frikken & Atallah 2003); cooperative scientific computation (Du & Atallah 2001; Goldwasser & Lindell 2002); on-line auctions (Cachin 1999); and secure comparison of information (Fagin, Naor & Winkler 1996). However, most of these protocols and solutions suffer from a possible large computational overhead resulting from use of public-key algorithms (Li et al. 2008). Thus, in early solutions, the computational complexity was exponential to the number of bits used to store the private records and to the number of records (Cachin 1999). However, Naor & Pinkas (Naor & Pinkas 2001) managed to lower the number of exponential operations required below one per record, by increasing the communicational complexity. Li and others (Li et al. 2008) discussed a system that does not make use of any public-key technology. In their system, Ex-OR operations are used to provide a different approach to Yao’s problem, although Impagliazzo & Rudich (Impagliazzo & Rudich 1989) argue that a system without a trapdoor function, such as public-key encryption, is unlikely to perform the required operations.

Most research into PIR and OT has focussed on perfectly developed schemes, with little attention paid to their practical use (Li et al. 2008; Naor & Pinkas 2001; Ostrovsky & III 2007; Wen-Guey 2002). In addition, none of the PIR/OT related research provides empirical evaluation. Comparison of the different schemes is usually done on the basis of computational and communicational complexity, which, some researchers assert (e.g. Li et al. 2008) should not be directly compared. Past research has shown that the efficiency of asymmetric key operations is 0.1% of the efficiency of symmetric key operations (Li et al. 2008). If this was true, a protocol that takes $O(n)$ symmetric operations, would take a similar amount of time to a protocol with $O(n / 1000)$ asymmetric operations. At the same time, computational complexity, expressed in terms of the number of operations undertaken, would suggest otherwise. Consequently, it could be argued that OT and PIR schemes are most practical when they are defined and evaluated for a specific problem, rather than for a general solution (Goldwasser 1997). This was also suggested by Naor & Pinkas (Naor & Pinkas 2001), who emphasise that selection of the trade-off between computational and communicational complexity depends on the specific problem at hand. Thus, in the following sections, a basic system for solving the privacy problem during inter-organisational and inter-departmental investigation is described, as well as an empirical proof of its efficiency for the task at hand.

4. Proposed System

The system for improving data subject privacy in investigations conducted by the public authorities is built on a base of theoretically tried-and-tested ideas and common encryption protocols currently in the public domain. The following section defines these building blocks, and the way they fit together.
4.1 Querying and narrowing the scope

The modern ISs may contain large numbers of records and only a narrow subset of such records would usually be related to a person being investigated, and therefore, be of interest to investigators. To safeguard privacy, the identity of the potential suspect cannot be provided to a data holder, a sender. Thus, the investigator, a chooser, needs to narrow down the scope of an inquiry, before privacy-preserving protocols may be used efficiently.

Depending on the level of privacy required, the scope of an enquiry should be narrowed to a number of records/identities between 100 and 1000, which this paper identifies as selected records. This may be achieved using techniques known from data mining. Thus, assuming that the chooser is a public authority that is investigating the data subject, it should be able to provide some basic information to narrow down the records to be processed. For example, the time of the last transaction performed by the suspect may be known by the chooser. This, in turn, can be used to limit the scope of the privacy-preserving enquiry to a set of data subjects who performed a transaction at around the same time as the suspect. The sender may be queried about its services usage statistics before the enquiry, in order to make sure the scope is not narrowed too much. For example, in a small on-line shop, the enquiry may be made about records of all customers that performed a transaction on a given date, while a larger e-commerce site may be queried about records of customers who performed a transaction in a much smaller time window, such as for less than one hour.

4.2 ID synchronisation

After the enquiry has been narrowed down, the chooser must check that the suspect’s data is among the selected records. The core of this system is a 1-n OT protocol that requires the chooser to know the index of the suspect’s data within the selected records, otherwise the protocol will not work. Thus, a commutative encryption is used for this purpose in the following way:

(1) The chooser encrypts the suspect’s ID with key \( E_{ch} \) and sends it to the sender.
(2) The sender encrypts the received input, \( E_{ch}(ID_{suspect}) \), with its own key \( E_{s} \), and sends \( E_{s}(E_{ch}(ID_{suspect})) \) back, together with all IDs in the selected records encrypted one-by-one, \( E_{s}(ID_{1}), ..., E_{s}(ID_{n}) \), where \( n \) is the number of the selected records.
(3) The chooser encrypts \( E_{s}(ID_{1}), ..., E_{s}(ID_{n}) \), the encrypted IDs received, with its own key, to produce \( E_{ch}(E_{s}(ID_{1})), ..., E_{ch}(E_{s}(ID_{n})) \) and compares resulting cipher-texts to the ID of the suspect encrypted by the both parties, \( E_{s}(E_{ch}(ID_{suspect})) \).

If any of the selected records refers to the suspect, its cipher-text will match \( E_{s}(E_{ch}(ID_{suspect})) \), and the chooser will recognise the index, \( i \), of the suspect’s data within selected records. At the same time, the sender will not be able to recognise the ID of the suspect, as long as the encryption used holds.

4.3 OT Protocol

ID synchronisation provides an index that is needed for the correct operation of 1-n protocol. This system uses a straightforward 1-n OT protocol, which is based on the 1-out-of-2 Oblivious Transfer (1-2 OT) scheme described in Schneier (1995). The resulting protocol operates as follows:

(1) The sender generates \( n \) sets of public/private keys pairs, and sends all public keys to the chooser, preserving the order in which they have been sent.
(2) The chooser generates a key with a private encryption algorithm, such as AES, later called AES key. It then uses the first public key received from the sender in Step 1 to encrypt the AES key and send it to the sender.
(3) The sender does not know which public key has been used to encode the AES key, or which record has been selected, thus protecting the privacy of the suspect. The sender can then decode cipher-text received in Step 2 using all private keys generated in Step 1, whilst preserving the order in which they have been decrypted. In this way \( n \) potential AES keys are created. Only the first one is the proper AES key; the other outputs are random sets of bits, which cannot be distinguished from ordinary AES keys.
(4) The sender encrypts all records using appropriate keys decrypted in Step 3. Thus, the first record in selected records is encrypted with an AES key decrypted using the first private key generated in Step 1. Consequently the first record, which includes data about the suspect, is encrypted using the AES key generated by the chooser in Step 2, sent to the sender encrypted by the first public key.
key, and then decrypted using $i^{th}$ private key. This way the $i^{th}$ record will be encrypted using the proper AES key.

(5) The chooser gets $n$ encrypted records, but using the AES key it is able to decrypt only the $i^{th}$ record. Other records are unreadable to the chooser provided that the false keys generated in Step 3, and used to encrypt these records in Step 4, are not broken.

4.4 Prototype

The feasibility of the above system was assessed with the use of a prototype, built in C# .NET and the Legion of the Bouncy Castle cryptography API (The Legion of the Bouncy Castle, 2007). To simplify operation and testing, the current prototype does not connect to a database, but instead uses files as records. Taking into consideration that IPv4 addresses are 32-bits in length, IPv6 addresses are 128-bits long, and a credit card number may be written using 54 bits, 256-bit identifier fields are used in the prototype. Such a solution provided sufficient data for the initial evaluation of the system, which did not, at this stage, include additional levels of complexity.

The first step is to narrow down the scope of the enquiry. For this operation, guidelines may be issued on the cooperation between the public authorities and data holders. However, due to large variations in ISs used by different organisations, this step cannot be automated. Thus, narrowing down the scope is not a part of the software prototype.

The ID synchronisation of the prototype uses a modification of the RSA scheme, sometimes referred to as SRA. This modification takes account of the fact that RSA is commutative for keys generated with common modulus $n$, and secure as long as the exponent is kept private (Chevalier, et al. 2005). During ID synchronisation, the chooser's key protects the identity of the suspect. Thus, different commutative encryption schemes, such as Pohlig-Helman protocol, may be used depending on the security requirements.

Finally, the 1-n OT protocol employed in the prototype uses RSA with 1024-bit keys and AES with 256-bit keys, which are currently standard in the encryption field. Breaking any or all of the $n$ RSA keys at this stage will not provide any measurable benefits to a potential perpetrator. However, if the private encryption scheme keys can be broken, the public authority, or an adversary (if the communication channel is not secured), could decrypt all selected records. Thus, it is important to choose a well-tested encryption scheme with an appropriate length of key.

5. Initial Evaluation

Usually protocols are evaluated on the basis of the number of operations and level of communication required. The ID synchronisation phase requires a maximum of $O(2(n+1))$ computations, but only two rounds of communication are needed. Likewise, the computational complexity of the 1-n OT used is large, and is affected by two major operations: symmetric key exchange; and processing of the selected results. Both of these operations require $O(n+1)$ each; however, only three rounds of communications are required, in total. Thus, the complete system requires only five rounds of communication, which simplifies the exchange of information between two parties. Consequently, real-time data exchange between the chooser and the sender is not necessary, so asynchronous transmissions may be used. This also permits the manual exchange of data, such as via post or secure email.

Figure 1 illustrates that the processing time is almost linear to the number of records in the set. Thus, processing time per record of a set-size can be established. For example, when running both the chooser and the sender processes on an ordinary workstation (RAM: 1GB, CPU: 1.6GHz), the total processing time is around 0.6 seconds per 1MB record. Increasing the physical size of records also affects the processing time in a linear manner (please see Figure 2 for details). More importantly, during the tests, CPU usage on the test workstations did not exceed 55%, meaning the impact of the investigative system on the underlying platform is limited.
The most costly operation in the whole system is the preparation phase. On the testing platform, generation of 100 RSA key pairs took around 160 seconds for 1024-bit keys, and around 85 seconds for 786-bit keys. This appears to be the only downfall of the system, however, the RSA keys may be prepared beforehand by the sender and stored in a repository until they are required. Thus, this aspect is not considered to be a major disruptive issue.

6. Conclusion

Modern research in multiparty computation is rich in theoretical solutions related to privacy concerns. Some of the solutions are mature enough to assist public authorities and internal investigators in safeguarding the privacy of the data subjects during inter-organisational and inter-departmental investigations. Arguably, before such solutions can be legally used in the inter-operational domain, some amendments to the current data protection laws will be required. The reason for this is the fact that 1-n OT protocols, apart from the requested \( R^i \) record, pass all other records to the requesting party. From a technical point-of-view, these records are encrypted in a way that renders them unusable. However, from a legal perspective, these records would be considered as processed data and as being sent to the requesting party unlawfully – that is, outwith the permitted exemptions allowed by the DPA.

The results of the evaluation clearly show that use of privacy-preserving systems during investigations is possible without causing delay to the investigations and with a negligible impact on the level of processing required by the third party providing the records. Operations requiring the input of such a third party, can be performed quickly on a standard developer’s or administrator’s workstation. This paper has proposed a system that would successfully protect Bob’s privacy in the scenario described in Section 1. Thus, the security services would obtain the data that would clear Bob from suspicion, without identifying him as a suspect to the on-line shop. The whole operation would take around 6 minutes if the records processed by the system could be narrowed to down to \( n \) equal 500.
This paper provides a description of a simple, but complete system that may be used for both inter-organisational and inter-departmental investigations. The simplicity of the design allows full comprehension of the operations performed, and understanding of the system’s potential. The transparency of the design is increased by the fact that well tested, widely available private and public encryption algorithms are used. Although C# was used for prototyping, a variety of different programming languages may be used to implement the real-life system. Moreover, two parties should be able to engage in protocol using different implementations of the system, since its specification will be in the public-domain. This system, apart from the small number of communication rounds required, has the advantage of good scalability, since the processing time is linear to the number of records processed and to the size of records. Thus, this paper could be regarded as a step forward in safeguarding the privacy of individuals during investigations. With such system in place, there is no need to compromise the privacy of suspects due to time restrictions. The time required is small compared to manual data handling methods, used currently to obtain a suspect’s data from third parties. Further planned research will involve making the above system more efficient without affecting its complexity, followed by more detailed evaluation using synthetic and real data.

7. Acknowledgements
The authors would like to thank Professor Rao Bhamidimarri, Dean of Faculty of Engineering and Computing, for making funding available for research, as well as Bob Rankin, Head of School, for assigning grants to this research program.

8. References


Abstract:

Privacy issues are becoming a key focus with software systems. Surveys show that the invasion of privacy is among the things people fear the most from the coming years. These fears seem to be justified, in the light of recent events involving the UK government. Thus, according to the EU Telecoms Commissioner the UK government breach European privacy laws by allowing a group of UK based Internet Service Providers (ISPs) to intercept communications of their users for behavioural advertising purposes. In this case it was complaints from the concerned public that made the EU Commission examine the privacy implications. Yet, on the contrary, popularity of various social networking portals, where users publish their personal and sensitive data publicly, is growing. Therefore, some argue that users should not expect any level of privacy in the digital world. Such climes are backed-up by the fact that majority of Internet users are unconcerned about the digital footprint they leave behind. What is overseen is the control factor. Users want to have the right to decide what information about their lives is in the public domain. Consequently, ‘one-size fits all’ solution to privacy concerns does not exist, as everybody perceives privacy in a slightly different way. Therefore, parties involved in data-handling, including social networking portals, need to research and implement privacy technologies that can keep their customers happy and make the operation comply with local security and privacy directives in many locations around the globe.

This paper gives an insight on how Privacy Enhancing Technologies (PETs) can be used to perform private matching operations in large datasets. These operations can be used by data-holders and individuals to compare or to retrieve information in a private manner in cases where trusted third party does not exist or trusted third party it used trusted for authentication purposes only. Thus, they can provide users with greater control over how their data is used. They include equality tests, dataset intersections, dataset equijoins, and symmetric private information retrieval protocols. Application of such private operations lies in the area of pervasive computing, database interaction, auditing and data acquisition. Here it is shown that PETs based on commutative cryptosystems are most efficient in performing these operations. Therefore, these cryptosystems are examined in detail. Currently anyone wishing to implement PETs based on commutative cryptosystems will quickly notice that such cryptosystems cannot be found in any of the popular cryptographic suites. The reason for this is the fact that these cryptographic algorithms are expensive to run in comparison with other encryption technologies and have limited area of usage in security applications. Thus, the key contribution of this paper is a guide to implementing commutative cryptosystems, using common open-source cryptographic packages. Consequently, this should enable developers and researchers to further investigate the existing PETs and propose new systems employing the notion of the commutative cryptography.

Keywords: commutative cryptography, data acquisition, privacy enhancing technologies, data mining, private matching

1. Introduction

In September 2000, a poll asking Americans what they fear the most in the upcoming century was published by the Wall Street Journal (in Swire & Steinfeld, 2002). The respondents were given a number of scenarios to list in order of most feared. These scenarios included international terrorism, global warming and world war. It turned out that among these the most feared was invasion of personal privacy. Things changed slightly after the 9/11, however, once enough time will pass from these tragic events privacy will, most likely, become a major concern once again. This trend is shown in the survey conducted by the Washington Post in 2006 (Balz & Deane, 2006), where 32% of respondents agreed that they would prefer the federal government to ensure that privacy rights are respected rather than investigating possible terrorism threats. This was 11% increase from the similar survey conducted in 2003.

European Commission has requested UK government to stop the Phorm programme, which analyses behaviour of internet users at ISP level for targeted advertising purposes (RAPID, 2009). This was a
response to a number of complaints by UK internet users received by the EU Telecoms Commissioner. Privacy activist groups welcomed this move by the European Commission, since the *Phorm* programme indulges into deep packet analysis of user traffic, which is equivalent to tapping into Internet conversations of every customer from participating ISPs. Consequently, one of the large on-line retailers in UK, Amazon, has decided to block *Phorm* from analysing traffic to and from their domains, and other businesses are likely to follow (Waters, 2009). This shows large groups of users are ready to fight their right for privacy, since it was their protests that triggered the actions of the EU Commission. However, Amazon itself tracks users in order to build their behavioural profiles, but it does it overtly and users can choose not to deal with this particular retailer. Thus, it can also be concluded, that it is the control element that is appreciated by the Internet users.

On contrary to the demands for privacy, a lot of the Internet users jeopardise their privacy and security by making their personal profiles openly available on social networking portals (Anderson, 2008). Some information posted on the Internet by unaware users is sensitive and can be used by adversary in identity theft and other social engineering exploits. However, this risk can be reduced by appropriate awareness campaigns and filtering mechanism integrated into web browsing or antivirus software. Where at the same time, relevant surveys show that users are becoming more relaxed about the *digital footprint*, i.e. traces of lives, they leave behind on the Internet (Stross, 2007). While this footprint can affect further lives of the users consciously choosing to publish their personal information on-line, it should remain their choice. Publishing information to social networking portals should not be treated as privacy or security risk, but more, as a lifestyle decision, as long as it is an informed decision.

The previous paragraphs show that users of software systems have various requirements as to their privacy. However, they all have one in common: all users want to maintain the control over the way their data is being used. In European Union appropriate legislations are there to safeguard this level of control over the private data. However, it is different in other parts of the world. Also, while the Western Society enjoys many liberties, people around the world still need to hide their beliefs and sexual preferences. Thus, there is a need for technologies that can allow like-minded people to find each other in the global village and communicate securely. In some cases soft security technologies can be used to improve privacy, these are policies and access control mechanisms that are more suitable than use of technologies supported by cryptography and number theory. However, these technologies can help only in the case where trusted third party exists. For example in the case of social networking portals soft security and training can be sufficient, since mutually trusted third party, the portal regulated by various privacy laws, is there to provide any level of privacy required by the users. However, the need for privacy enhancing solutions based on cryptography is still there. When parties cannot agree upon a trusted third party, or there is a need for secrecy of communication, mathematics can become a trustee of the secrets. In this paper following scenarios are used to help illustrate the need for private operations on datasets:

**Scenario 1:**
Pervasive computing enables electronic devices to assist people in their everyday lives, even in the most intimate scenarios. However, in a distributed network of pervasive computing devices, a common trusted third party does not always exist. Imagine avatars running on PDAs and mobile phones that scan the surroundings in search of an ideal match for a date or a social networking contact. Such avatars, based on their owner’s profile, could compare profiles supplied by avatars of other users nearby. A simple solution would provide profiles openly to others users of this dating or social networking application. Nevertheless, this could make the user vulnerable to many well targeted scams and attacks. With the use of PETs described in this paper the profiles could be confidential and still the avatars would be capable of comparing them.

**Scenario 2:**
Security services investigate a terrorist suspect, Bob. They need to obtain the information on his recent Internet activity. Currently, in the UK the procedure for obtaining such data involves serving a notice to the ISP of the suspect. This notice would need to identify Bob as a suspect, which would breach Bob’s privacy and possibly jeopardise the investigation (Kwecka, Buchanan, Spiers, & Saliou, 2008). On the other hand, it is possible to build a system based on PETs that would allow the ISP to give access to Bob’s data, without security services having to reveal his identity, and without breaching the privacy of the other data subjects in the ISP’s database.
This research claims that PETs can assist in the solution to a number of private matching problems similar to those described by the scenarios. It shows that use of commutative cryptosystems is beneficial to this task and then provides a guide to implementing secure Commutative Encryption (CE) protocols.

2. Commutative Encryption

This research argues that private matching operations of datasets are most efficient when based on CE. The reasoning behind this claim is provided in Section 3, while this section discusses the principals of operation of CE protocols. Thus, in conventional cryptography when a plaintext message is encrypted with two different cryptographic functions $E_A$ and $E_B$, the resulting ciphertext will be different depending on the order of the key application. Thus, for two different conventional cryptographic functions $E_A$ and $E_B$, and for any value in the allowed input range following is true:

$$E_A(E_B(value)) \neq E_B(E_A(value))$$

for conventional cryptography

For most cryptographic application, this is desirable behaviour, since it improves the security of the plaintext. However, most protocols described in Section 3 require the underlying cryptographic protocol where opposite of Equation 1 is true:

$$E_A(E_B(value)) = E_B(E_A(value))$$

for commutative cryptography

This requirement is met only by commutative protocols. Consequently, such properties make CE protocols an interesting choice for private matching algorithms.

3. Private matching dataset operations

The privacy concerns described in Section 1 fall into a general area of private matching techniques. This problem should not be confused with statistical data mining, where efforts are focussed on providing privacy in systems that share data for the purpose of statistical analysis. Thus, in the later problem anonymisation techniques can be successfully deployed to guard privacy of the records, however, these techniques cannot assist in private matching problem. In this section few privacy-preserving protocols allowing data comparison are described.

3.1 Private Equality Test (PEqT)

A key primitive in the area of private dataset operations is PEqT that allows two parties to test their secret inputs for equality. One of the first such tests published was that described by Frikken & Atallah (2003) and shown in Figure 3.1. This scheme was designed to work with CE protocols and the operation of the protocol can be summarised as follows, for two parties, Alice and Bob, each holding different CE keys:

1. Alice encrypts her input and sends it to Bob,
2. Bob encrypts the ciphertext received from Alice and sends it back,
3. Bob encrypts his secret input and sends it to Alice,
4. Alice encrypts the ciphertext containing Bob's input,
5. Alice compares the two resulting ciphertexts, if they are equal then her and Bob's inputs are equal,
6. Alice may inform Bob about the result.

The above PEqT is the first solution to the problem of comparing information without leaking it that does not require trusted third party. In 1996 Fagin, Naor and Winkler have collected and published a number of real-life substitute solutions to this problem, however, most of the proposed solutions which did not require trusted third party involved Bob and Alice to monitor each other. Therefore, these solutions were not real alternative to CE PEqT. To this date, a number of different PEqT schemes have been proposed, however, complexity of the other schemes is usually higher than this of CE solution. Boa and Deng (2001) proposed quite an efficient method for equality testing based on homomorphic encryption. However, this method requires a series of multiplications, an exponentiation, as well as a round of homomorphic encryption and decryption. The homomorphic protocol proposed for their scheme is ElGamal (EG), which itself requires two exponentiations modulo
a prime during the encryption process and another one for the decryption operation. Consequently, complexity of their protocol is slightly higher than complexity of the CE PEqT scheme illustrated in Figure 3.1 when it is based on Pohlig-Hellman (PH) algorithm described in Section 4. However, only the slight difference in performance means that the decision of using one or the other method should be based on factors other than efficiency alone.

![Figure 3.1 Private Equality Test](image)

**3.2 1-to-n Private Equality Test**

It is rare for two parties to want to compare only a single value. More often, there are a number of inputs to be compared. Unfortunately, the complexity of most PEqT schemes cannot be lowered in case there are more inputs to the protocol. This applies also to the homomorphic scheme designed by Boa and Deng (2001). In the case of their protocol the whole PEqT would need to be run \( n \) times in order for Alice to compare her input to \( n \) inputs held by Bob. Therefore, such operation would require \( O(4n) \) exponentiations in total. This is not the case with a slight modification of the CE based PEqT. Here Bob’s inputs must be encrypted only once, using a single exponentiation operation, and Alice single input needs to be encrypted by her and Bob, and later decrypted by her. Thus, in total CE PEqT scheme requires \( O(n+3) \) exponentiations.

The CE PEqT protocol for Alice to compare her secret input with \( n \) inputs held by Bob involves the following steps:

1. Alice encrypts her input and sends it to Bob,
2. Bob encrypts the ciphertext received from Alice and sends it back,
3. Alice decrypts the ciphertext containing her input encrypted by her and Bob,
4. Bob encrypts all his secret inputs and sends them to Alice,
5. Alice compares the result from Step 3 with other ciphertexts received from Bob in Step 4, and if equal ciphertext are found, then she knows that Bob has got a common element with her set,
6. Alice may inform Bob about the result.

Figure 3.2 illustrates this protocol.
As far as the openly available literature goes the above protocol is the most efficient 1-to-n PEqT protocol available. It can be used as a partial solution to the problem posed by the two scenarios provided in Section 1. Thus, in Scenario 1 an avatar belonging to Alice could communicate with other avatars and check do their owners like “music”, “horror movies” or “quantum physics”. However, each of these queries would require a separate run of the above protocol. In Scenario 2 the security services could use the protocol to find out is Bob a client of a given ISP, or index of Bob’s records in the ISP’s database, but they would be unable to request any more information about him in a private manner.

3.3 Private intersection, private intersection size

Section 3.1 concluded that PEqT can be efficiently performed using CE. Agrawal, Evfimievski and Srikant (2003) extended the above 1-to-n PEqT algorithm to construct a private intersection and private intersection size protocols. The operation of their private intersection size protocol can be described as follows:

1. Alice and Bob apply hash function to their sets.
2. Both parties encrypt their hashed sets using their CE encryption keys.
3. Alice sends Bob her encrypted set in a lexicographical order.
4. Bob encrypts the set received from Alice in Step 3, and sends back the resulting ciphertexts reordered lexicographically.
5. Bob sends Alice his encrypted set in a lexicographical order.
6. Alice encrypts the set received from Bob in Step 5.
7. Alice compares the two sets, i.e. the one received in Steps 4 and the one produced in Step 6. The elements from one set that have a match in the other form the intersection of the sets, hence the count of the common elements is the intersection size.
8. Alice may inform Bob about the results.

Notice that in the above protocol Alice and Bob are unable to find out which elements are common. This protocol provides them only with the information about the size of the intersection, but it can be easily transformed into private intersection protocol. If the parties would like to identify which elements are common between them, then Bob would have to pair up inputs from Alice, with his encryption of these inputs in Step 4. This way, in Step 7 Alice could find out which of the elements in her set match Bob’s elements. Agrawal et. al. provide strong proofs of correctness and security of the above protocols.
Both private intersection size and private intersection protocols can be used to create proximity based social networking or dating application described in Scenario 1. Avatars could scan surroundings for other avatars, and then engage in the above protocols to find out how many things their users have in common or what exactly they have in common. No other information would be revealed. Consequently, this is a complete solution to the problems presented by Scenario 1. On the contrary, these protocols have limited benefits to Scenario 2, since they can help the security services to search for their suspect in the ISP’s database, but they don’t provide any information retrieval mechanism. Thus, the security services would be unable to gather any data about their suspects.

### 3.4 Three-Pass (3Pass) protocol

Before operation of other private dataset protocols can be described 3Pass protocol primitive needs to be explained. This protocol was intended so that two parties could share a secret without exchanging any private or public key. Thus, the protocol was aimed at providing an alternative to public-key encryption and DH-like key negotiation protocols. The 3Pass, though, have never been widely used in this way since, on its own, it is susceptible to the man-in-the-middle attacks (Shamir, Rivest, & Adleman, 1981) and is less efficient than Rivest-Shamir-Adleman (RSA), a common choice public-key algorithm (Rivest, Shamir, & Adleman, 1978).

![3Pass protocol diagram](image)

**Figure 3.3** Operation of the 3Pass protocol

The 3Pass protocol can be described using the following physical analogy:

1. Alice places a secret message in a box and locks it with a padlock,
2. The box is sent to Bob, who adds his padlock to the latch, and sent the box back to Alice,
3. Alice removes her padlock and passes the box back to Bob,
4. Bob removes his padlock, and this enables him to read the message inside the box.

A more formal notation of this protocol is shown in Figure 3.3. Using this protocol Alice and Bob can share a secret without sharing a key. This algorithm needs the underlying encryption to be commutative. The CE protocols allow for a ciphertext created using a number of keys to be decrypted with application of the decryption keys in an arbitrary order. This is not the case in non-commutative ciphers and a ciphertext message created using non-commutative protocols can only be decrypted in a reverse order to the encryption process that produced it, otherwise the decryption output would be gibberish, and not the plaintext message.

### 3.5 1-out-of-n Oblivious Transfer (OT)

To OT protocols allow a client, often referred to as a chooser, to retrieve a record from a number of records hosted on a server, usually called a sender, in a way that the chooser gets only the record requested and the sender learns nothing apart from the fact that the chooser retrieved a record (Schneier, 1996). Generally, the chooser needs to provide the sender with an index i of the record that needs to be retrieved. This selection is often random or based on publicly available catalogue of items held by the sender. Thus, an OT algorithm could be used for a purchase of digital goods in a
way that the provider cannot identify the product purchased by the buyer, while the protocol guarantees that the buyer can only get access to one product. This is possible, since in this scenario the catalogue of products can be made public. Such solution was proposed by Bao and Deng in 2001, who designed an OT based on the 3Pass protocol. Many other efficient 1-out-of-n OT algorithms exist, such as those described in Lipmaa (2003) and Aiello, Ishai, & Reingold (2001). On the contrary, OT on its own is not a viable solution to the privacy concerns described in the scenarios provided in Section 1. It has no use in Scenario 1, since the avatars are looking to compare data and not to retrieve a record. Also, it cannot be used in Scenario 2. Although, the OT primitive allows a retrieval of a record, there is no way for the security services to provide the index $i$ of the ISP’s dataset that would relate to their suspect.

The OT primitive needs to be combined with a form of PEqT, such as 1-to-n PEqT or private intersection protocol in order to allow the chooser to identify an index of the interesting record in the data-holder’s set. Such a composite solution was presented by Kwecka, Buchanan, Spiers, & Saliou (2008). The authors proposed a system that would be suitable, for security services from Scenario 2, to acquire a record from the Bob’s ISP, maintaining the privacy of the suspect, other record, in the ISP’s database, and also privacy of the investigation itself. However, this solution is not scalable, as for each suspect police would have to run a separate 1-to-n PEqT and OT enquiries.

### 3.6 Private equijoin

The previous paragraph suggested that it is possible to create a viable solution to the problem presented in Scenario 2 from a mixture of PEqT and OT protocols. Such solutions usually do not blend well, and usually each requires its own computationally expensive preparation phase. However, Agrawal et. al. (2003) noticed that if CE is used for both PEqT and OT, an efficient protocol for private data acquisition can be formed. Thus, when the private intersection protocol described above is combined with the 3Pass primitive an ideal solution to Scenario 2 is created. The security services can retrieve a number of records relating their suspect with a single run of a protocol. Resulting protocol is referred to as a private equijoin protocol and is an extension of $k$-out-of-$n$ OT protocol, where $k$ records can to be retrieved over a single OT run. In such protocol the chooser, learns some extra information from the sender for the elements which form the intersection between the sets held by the two parties.

### 4. Implementing commutative cryptosystems

At first the only cryptographically strong commutative protocols were the ones based on modular exponentiation (Shamir, 1980). However, most theoretic applications of CE could benefit from being based on more efficient symmetric algorithms. Unfortunately, with the exception of the stream ciphers, which are unusable for dataset operations, all the current implementations of CE are still based on modular exponentiation, quite an inefficient process, often used in asymmetric encryption. This section explains the choice of CE to support the protocols described in Section 3 and provides description of implementation of the most viable solution.

#### 4.1 Commutative algorithms

In his PhD thesis Weis proposed a method of building a commutative cryptosystem from an arbitrary cryptosystem that supports homomorphic multiplication of ciphertexts (Weis, 2006). Weis provides an example based on EG protocol that uses the original EG key specification and generation. However, his algorithm displays only one of two commutative properties that are required for most protocols discussed in Section 3, since Equation 2 is not true for this algorithm. Consequently, it can be used in 3Pass-like protocols but not in the protocols that allow comparison of values, through comparison of ciphertexts.

The only protocol viable to perform PEqT, private equijoin and other protocols needed for the scenarios described in Section 1 is in fact an unpopular PH encryption scheme that is based on exponentiation modulo a large prime $p$ (Pohlig & Hellman, 1978). This algorithm is commutative for keys with common prime $p$ (Shamir, 1980; Shamir, Rivest, & Adleman, 1981). Interestingly, the PH algorithm, although similar to RSA algorithm, does not support public-key operations, since in PH the decryption key can be easily calculated from the encryption key. Nor it is a symmetric algorithm, as two different keys are used for encryption and decryption. Therefore, PH can be considered as an asymmetric private-key encryption algorithm. This is not a popular type of cryptographic protocol, and this explains why PH cannot be found in any openly available cryptographic suite.
4.2 Commutative PH algorithm
The PH scheme is similar to RSA. Equation 3 and Equation 4 show PH encryption and decryption functions respectively, where \( M \) stands for the plaintext message and \( C \) stands for the resulting ciphertext:

\[
\text{Equation 3: } C = M^e \mod p
\]
\[
\text{Equation 4: } M = C^d \mod p
\]

Both operations are performed modulo a large prime \( p \), and different keys, exponents, are used for encryption (exponent \( e \)) and decryption (exponent \( d \)) in this algorithm. The encryption exponent \( e \) is randomly chosen so that:

\[
\text{Equation 5 } 1 < e < p - 1
\]

Then, \( e \) is used to calculate the decryption exponent:

\[
\text{Equation 6 } de \equiv 1 \mod (p - 1) \iff d = e^{-1} \mod (p - 1)
\]

Unlike RSA, it is easy to calculate the decryption key from the encryption key, thus, \( e \) must remain secret.

4.3 Security of PH algorithm
In PH both, encryption and decryption, keys should be kept secret. However, there is no harm in making the large prime \( p \) public, while this is required in order to make PH scheme commutative. An adversary with the knowledge of the ciphertext \( C \) and the prime \( p \) would need to solve the following hard problem to break the commutative PH protocol (Schneier, 1996):

\[
\text{Equation 7 } e = \log_p C \mod p
\]

Just like RSA, the ciphertext created using the PH protocol may leak some information about the input plaintext message. Therefore, this algorithm is mainly suitable for situations where the input is formed from random data, such as secret encryption or hashed signatures. It is also advised to use padding schemes together with any RSA implementations (Kaliski & Rivest, 2003), consequently implementations of PH should employ a padding scheme as well. Therefore, if PH algorithm is to be employed in implementing protocols described in Section 3, then the input data can be hashed to form blocks of data that would not require padding (Agrawal, Evfimievski, & Srikant, 2003).

These issues show that implementation of the PH should not be treated lightly. The operations involved in the protocol itself can be easily performed using any package enabling operations on large integers. However, there are other components to the protocol, such as hashing and random number generation. Many tried-and-tested cryptographic suites exist to support these operations. Thus, for security reasons, it is recommended to build the PH algorithm into an existing cryptographic suite that allows modification of its source code. Since, PH is similar to RSA, in operation, an RSA implementation can be easily altered to support PH algorithm and the transformations required are shown next.

4.4 RSA based implementation of PH
Parties wanting to employ commutative behaviour of PH must use a common \( p \). Consequently, \( p \) can be decided upon during initialisation phase by the parties involved in the protocol, or it can be written into protocol specification. It is advisable to use a strong prime for PH, just like it is in case of DH algorithm (Agrawal, Evfimievski, & Srikant, 2003; Ferguson & Schneier, 2003; Silverman & Rivest, 2001). A prime \( p \) is strong if there is such a prime \( q \) that the following is true:

\[
\text{Equation 8 } p = 2q + 1
\]
Most cryptographic suites have build-in methods for generating strong primes, often hidden in DH key generator code, this methods should be used for obtaining \( p \) (and \( q \)) for this PH implementation. Once \( p \) is agreed upon the PH key generation may begin.

As previously mentioned RSA implementations can be used to implement PH algorithm. The main difference between the two is that, in RSA, encryption and decryption are performed modulo a product of two large primes \( p \) and \( q \), whereas in PH these operations are performed modulo a prime \( p \). Thus, there are no changes required to the actual RSA engine, since its main task is to perform exponentiation modulo a number. However, there is a slight difference in the key generation phases of these two algorithms. It has more to do with the way most RSA key generators have been implemented, rather than the differences in algorithms themselves. Some RSA implementations available in the public domain do not choose \( e \) randomly, but use values 3 or 17 just as long as \( d \) generated from \( e, p \) and \( q \) is large enough for the RSA Problem to hold. Using small exponent \( e \) speeds up key generation and encryption, but it could make RSA implementations employing this technique weaker (Kaliski & Rivest, 2003). In PH, knowing the encryption exponent \( e \) and the prime \( p \) is enough to calculate the decryption key \( d \). Since \( p \) is public, \( e \) needs to be kept private. Consequently, using small values for \( e \) would break this cryptosystem, and so an ordinary RSA key generator is unsuitable to produce valid PH keys.

The RSA key generator needs to be altered so that it accepts \( p \) (and \( q \)) as an input, and generates random \( e \) and then calculates \( d \). The exponent \( e \) should be randomly chosen odd integer from the range defined in Equation 5 (Bao & Deng, 2001) and the Greatest Common Divisor (GCD) of \( e \) and \( (p-1) \) should be one, i.e. \( e \) needs to be relatively prime with \( (p-1) \). Once the appropriate \( e \) is generated the decryption exponent \( d \) can be calculated according to Equation 6. This is usually easily done in cryptographic suites, since methods for calculating modulo inverse of a number are used in RSA key generation. After such modifications the PH protocol can be invoked just like an ordinary RSA algorithm would be invoked.

4.5 Licensing

Both PH and RSA were patented in early 80’s, consequently these patents have now expired and the above solution can be used without a licence (Schneier, 1996). However, attention should be given to the licence of the cryptographic suite that is being modified with PH scheme, since licences of some open-source packages prohibit source-code modification.

5. Conclusions

This paper has reviewed and analysed applications of PETs to privacy problems in matching operations on datasets. Section 1 provided usage scenarios for such operations. The findings show that protocols based on commutative cryptosystems are most viable private matching algorithms, when efficiency is the key consideration. This is the case when more than one value is to be compared, since other efficient privacy enhancing algorithms to perform a single value equality test exist.

To answer the issues raised by Scenario 1. Private intersection and private intersection size protocols described in Section 3 can be used in pervasive computing applications where avatars scan surrounding of their owner for services and other avatars that match certain selection criteria that should remind secret in order to maintain the privacy of their owner. Thus, the future users of proximity based social networking and dating application can be more protected against various social engineering attacks and targeted advertising campaigns if these protocols are deployed.

Scenario 2 describes a case where secret service’s investigation can be jeopardised together with suspect’s privacy during a lawful communication data acquisition process in the UK. In Section 3 it was shown that these concerns can be mitigated by a composite system build from different 1-to-n PEqT and OT protocols. However, such solutions are usually inefficient, since both 1-to-n PEqT and OT protocols require certain preparation phase before the protocols can run. Also, these processes are often expensive in terms of computation needed. Therefore, a complete scheme based on commutative cryptography, i.e. private equijoin protocol by Agrawal et. al. (2003), is suggested for performing symmetric private information retrieval tasks.
Protocols described in Section 3 can form a key part of a number of PETs, where there is not trusted third party, or such party is trusted only for authentication purposes but not to handle sensitive data. These protocols are most efficient when based on commutative cryptosystems. However, CE cannot be performed using the openly available cryptographic suites, and thus, fast development strategy for forming secure implementations of CE is required. This paper presents a method of modifying common implementations of the RSA scheme, so that PH algorithm with commutative properties can be achieved as a part of any cryptographic suite that allows modification of the source-code. The reason for implementing PH algorithm as a part of a cryptographic suite is that it should always be used in conjunction with hashing or padding schemes and well tested random number generators. Such solution allows building a complete infrastructure for private matching in most secure and cost effective way.

This research focused on private matching dataset operations and showed that there is a large field of use for the presented algorithms. However, they have certain limitations. In all the protocols described here the match must be perfect. Thus, it is not possible to match fuzzy numbers and patterns. For example the techniques described could not be used to create a system for private DNA searches, since the object of the comparison in DNS analysis is a pattern and not a set value. Currently there is no way to compare patterns or fuzzy numbers in a private manner, and thus, this is the area of the further research in private matching algorithms.

6. Acknowledgments

The authors would like to thank Dr Sandra Cairncross, Dean of Faculty of Engineering, Computing and Creative Industries, for making funding available for research, as well as Sally Smith, Head of School, for assigning grants to this research program.

7. References


Privacy-Preserving Data Acquisition Protocol

Zbigniew Kwecka*, Prof. William J Buchanan† and Duncan Spiers†

* Edinburgh Napier University/Centre for Distributed Computing & Security, Edinburgh, UK, z.kwecka@napier.ac.uk
† Edinburgh Napier University/ Centre for Law, Edinburgh, UK

Abstract—Current investigative data acquisition techniques often breach human and natural rights of the data subject and can jeopardize an investigation. Often the investigators need to reveal to the data controller precise details of their suspect’s identity or suspect’s profile. In this research a novel approach to investigative data acquisition is presented and privacy preserving Investigative Data Acquisition Protocol (IDAP) is defined. This protocol is the first that allows for performing private information retrieval of records matching multiple selection criteria.

Index Terms—cryptography, data acquisition, privacy.

I. INTRODUCTION

Surveys show that the invasion of privacy is among the things people fear the most from the coming years [1][2]. Emerging technologies allow for fast digitalization of operational procedures in many organizations, and depending on how these technologies are used the result can be destructive or beneficial to privacy of the parties involved.

The public authorities such as Police, Customs, and Tax Offices need to request information from third-parties on regular basis and the data protection legislations allow for such requests even without warrants [3][4]. Depending on the way these requests are performed human and natural rights of the data-subject can be breached and/or investigation can be jeopardized [5]. This research gives an insight on how the Privacy Enhancing Technologies (PETs) can be used to improve investigative data acquisition techniques in order to protect the data-subjects and the secrecy of the investigations. It defines a novel, efficient approach to maintain the secrecy of an enquiry.

Motivating scenario: Police has a profile of a suspect (e.g. sex, age, and ethnic origin) and would like to find individuals fitting this profile working in organizations in a neighborhood to the crime scene, but revealing the profile to these organizations may harm the investigation and individuals matching the profile.

Currently the police would often have to delay their enquiries in order to protect the investigation as well as the innocent individuals fitting their profile. For example if the case being investigated had a public tension around it, and the suspect’s profile matched individuals in a local minority, the enquiry could have serious consequences to the members of this minority.

To generalize the problem the party making the enquiry, the chooser, can request $k$ parameters, referred to as return parameters $r_{pi}$ from a source table or a view held by the data holder, referred to as the sender. While doing so, the chooser may wish to keep the values $x_{i,j}$ of the $i$ input parameters $ip_{i,j}$ secret. Such a query can be mapped into an SQL query as follows:

$$\text{SELECT } r_{p1}, r_{p2}, \ldots, r_{pk} \text{ FROM source}$$
$$\text{WHERE } ip_{1}=x_{1} \text{ AND } ip_{2}=x_{2} \text{ AND } \ldots \text{ AND } ip_{l}=x_{l}$$

This research proposes a novel protocol that could solve this problem using a combination of commutative data locking based on a well-known three-pass (3Pass) secret exchange protocol discussed by Shamir in [6][7] and the Private Equijoin (PE) protocol defined in [8].

II. BACKGROUND AND RELATED WORK

The problem of retrieving data records in a private manner from a database hosted by a third-party is not new. Thus, there are a number of primitives that can be used to approach such problem. First there was a Private Information Retrieval (PIR) primitive where a chooser could query a database in a way that sender is unable to identify which row of data is being retrieved. The main motivation behind the PIR schemes is the achievement of as low communicational and computational complexity as possible, where the privacy of the records in the database is not a concern [9]. A stronger notion than PIR is $1$-out-of-$n$ Oblivious Transfer (OT) primitive that allows the retrieval of a randomly selected record from the dataset of $n$ elements held by sender, in a way that the sender cannot learn which record has been transferred, and the chooser cannot learn anything about other records in the dataset [10]. $1$-out-of-$n$ OT protocols that allow chooser to actively select a record to be retrieved, and that have linear or sub-linear complexity, are referred to as symmetric PIR protocols. These are the most useful privacy-preserving data retrieval protocols, and find use in electronic watch-lists of suspects [11]; cooperative scientific computation [12][13]; and on-line auctions [14].

The information retrieval protocols described above are capable of collecting a record, or records, from a specific index in the sender’s dataset, but they do not support a search or comparison functionality that could be used to perform an investigation of the dataset. Instead it is often assumed that some indexed description of data is publically available in a
form of electronic catalogue [15][16]. Consequently if a system is to allow data retrieval based on private selection, it must implement some private comparison function. There are a number of protocols that allow two parties to compare their values in a private manner, i.e. to compare information without leaking it. But only some are optimized to make comparisons for equality, and these are referred to Private Equality Test (PEqT) protocols. PEqT protocols are often based on commutative [5][11] or homomorphic cryptosystems [15]. Often two different protocols, each with separate and computationally expensive preparation phases need to be used to first make a selection and then retrieve a selected record. The exception to this rule is a range of protocols including: private intersection; private intersection size; and PE defined in [8]. For this reason the primitives in [8] are extended in this research to a setting where multiple matches on a given data row must be made in order to retrieve it and unlock it.

The following section describes the building blocks of the investigative data retrieval protocol defined in Section IV. These building blocks are a 3Pass primitive based on commutative encryption and the PE protocol proposed in [8]

III. BUILDING BLOCKS

A. Commutative Cryptosystems

Many cryptographic applications employ sequential encryption and decryption operations under one or more underlying cryptosystems. The reasons to sequence (cascade) different cryptographic schemes together include, strengthening the resulting ciphertext and achieving additional functionality which is impossible under any given encryption scheme on its own [17][18]. A basic cascadable cryptosystem can consist of a number of encryption stages, where the output from one stage is treated as an input to another. In such a basic cascadable cryptosystem it is necessary to decrypt in the reverse order of encryption operations. However, a special class of sequential cryptosystems - commutative cascadable cryptosystems – allows for the decryption of a ciphertext in an arbitrary order. Thus, a ciphertext \( c = e_a e_b (m) \) (\( e \) – ciphertext, \( m \) – plaintext, \( e \) – encryption operation under keys \( a \) and \( b \)), could be decrypted as either \( m = d_a d_b (c) \) or \( m = d_b d_a (c) \).

The advantages of such cryptosystems were widely promoted by Shamir in [6] as used in his, Rivest’s and Aldman’s, now classic, game of mental poker, employing the 3Pass secret exchange protocol.

This research uses and extends the primitive of the 3Pass protocol in order to lock query results to a set of query variables, as described in Section IV. The 3Pass protocol, shown in Fig. 1, was intended so that two parties could share a secret without exchanging any private or public key.

![Fig. 1. Three-Pass Secret Exchange Protocol.](image)

The operation of the protocol can be described using the following physical analogy:

1. Alice places a secret message \( m \) in a box and locks it with a padlock \( E_A \).
2. The box is sent to Bob, who adds his padlock \( E_B \) to the latch, and sends the box back to Alice.
3. Alice removes her padlock and passes the box back to Bob.
4. Bob removes his padlock, and this enables him to read the message from inside the box.

There could be more parties, or encryption stages, involved in a 3Pass-like protocol. In fact Khayat [19] formalizes this idea into a protocol for sharing a secret with a board of trustees. This protocol, itself, lacked the safeguards needed for a protocol intended for storing information securely for a number of years, since it fails if one of the parties has left the protocol without decrypting the ciphertext. Nevertheless such protocol is ideal for locking a plaintext multiple times and then unlocking it in an arbitrary order, as long as the parties are cooperating until the execution of the protocol is completed. Such functionality is required of the data acquisition protocol described later in this paper.

The first known commutative cryptosystem, that could be used to implement the 3Pass protocol, was based on the Pohling-Hellman (PH, asymmetric private key scheme [20]. PH was never a popular protocol, since being asymmetric algorithm it was slow in comparison to other private key encryption algorithms. However, the ever-popular, and still the most widely used, Rivest-Shamir-Adleman (RSA) public key cryptosystem is in fact based on the PH algorithm [21], and the RSA cryptographic engine is capable of performing PH encryption and decryption operations. The only difference between these cryptosystems is the fact that operations in PH are performed modulo a single large prime number \( p \), whereas RSA has a modulus made up of two large primes. Equations (2) and (3) show encryption and decryption operations under PH respectively. Encryption exponent/key \( e \) is selected at random, and the decryption exponent \( d \) is derived from the modulo-inverse of \( e \).

\[
c = m^e \mod p
\]

(2)
\[ m = c^d \mod p \]  
\[ e = \log_p c \mod p \]  

In order for the protocol to remain secure both \( e \) and \( d \) need to be kept secret. The protocol is commutative for operations modulo a given prime, thus all the parties in the system would have to know \( p \). Security is maintained since an adversary with the knowledge of the ciphertext \( c \) and the prime \( p \) would need to solve the hard problem (4) in order to determine one of the keys used \[10\]. However, the PH algorithm shares some weaknesses with its public key successor RSA, and it should only be used to encrypt randomized inputs, such as symmetric encryption keys or hash signatures, and a padding scheme should also be used \[22\].

When PH encryption scheme is used for cascascable commutative cryptographic operations, it is impossible to tell how many times the plaintext has been locked (encrypted by any party) and whether an operation has worked or not. The ciphertext can be encrypted and decrypted any number of times and there is no way to tell once the plaintext, usually a random number itself, has been reached. Thus, if only two parties are involved in a protocol, and they use one key each, these concerns can be negligible. However, if more parties are engaged in the protocol, such as in the sharing a secret with a board of trustees protocol by Khayat \[19\], or more than one key is used by any of the parties, such as in the data acquisition protocol proposed in this paper, this can become a concern. This, though, is addressed by Weis in \[18\] where he describes a method of building a semantically secure commutative cryptosystem from an arbitrary cryptosystem that supports homomorphic multiplication of ciphertexts. Weis provides an example based on ElGamal \[EG\], homomorphic cryptosystem that uses Diffie-Hellman (DH) key exchange algorithm's foundations. It uses original EG key specification and generation, i.e. first a strong prime \( p \) and generator \( g \) of the group \( \mathbb{Z}_p^* \) are chosen, then a private key \( x \) is drawn at random, and the public key \( y \) is calculated from \( x \), \( g \) and \( p \). For EG the ciphertext is made of two tuples \( \alpha \) and \( \beta \) (5), where \( r \) is a random number, whereas the commutative EG by Weis (CEG) forms the initial ciphertext in a similar manner, but it is made up of four tuples \( \alpha, \beta, \gamma, \delta \). The first two tuples \( (\alpha, \beta) \) form the actual ciphertext, while the other two \( (\gamma, \delta) \) are used as a checksum. The calculations involved in creation of the initial ciphertext in CEG, where \( r \) and \( s \) are random numbers, are shown in (6). The input to the CEG engine can be a secret message \( m \), for the initial encryption of the plaintext \( m \), or \( n \) different 4-tuple ciphertexts in the form \((\alpha, \beta);(\gamma, \delta)\), when the plaintext has already been encrypted \( n \)-times. In the second case \((n+1)\) values \( k \) are chosen so that their product is unity, and tuple \( \alpha \) of each input ciphertext is multiplied with a distinct value \( k \). Finally the sole remaining value \( k \) is encrypted just like \( m \) in (6). As the last step of the encryption protocol, all \( n \) input ciphertexts are re-encrypted (7), where \((\alpha', \beta');(\gamma', \delta')\) is the result of re-encrypting \((c\alpha, d\beta);(c\gamma', d\delta')\).

\[ c = (\alpha, \beta) = (y^m, g^r) \]  
\[ c = (\alpha, \beta);(\gamma, \delta) = (y^m, g^r);(y^r, g^s) \]  
\[ c = (\alpha', \beta');(\gamma', \delta') = (c\alpha, d\beta);(c\gamma', d\delta') \]

The key improvements of CEH over PH is the fact that the ciphertexts contain checksums and a party requested to perform a decryption can identify that the ciphertext received has or has not been previously encrypted by that party. This is done by searching the ciphertext for a four-tuple element \( c_0 \) such that \( (\gamma / \delta_1) = 1 \), where \( x \) is the private key of that party. If such an element exists, the ciphertext has indeed been encrypted by this party, and four-tuple \( c_0 \) is the element (lock) previously added to the ciphertext by that party.

When choosing whether to apply PH or CEG, dissimilarity between the cryptosystems should be noted. Ciphertext formed under PH following a number of encryption operations will be equal to a ciphertext formed using the same encryption operations in an arbitrary order. This property is required by the 3Pass protocol, whereas CEG does not have this property, and thus it cannot be used in the protocol described Section B or the 3Pass.

**B. Private Equijoin Protocol**

The most scalable and flexible of PEqT protocols for operations on datasets is the Private Equijoin (PE) protocol proposed in \[8\] by Agrawal, Evfimievski and Srikant. This scheme extended the classical commutative PEqT shown in Fig. 2, to allow conditional retrieval of data for the records that are common for both parties participating in a \( m \)-out-of-\( n \) PEqT protocol.

![Fig. 2. Private Equality Test.](image)

This protocol allows two parties to compare their secret inputs.

The operation of the PEqT protocol shown in Fig. 2 can be described in the following steps:

1. Alice encrypts her input and sends it to Bob.
2. Bob encrypts the ciphertext received from Alice and sends it back.
3. Bob encrypts his secret input and sends it to Alice.
4. Alice encrypts the ciphertext containing Bob’s input.
5. Alice compares the two resulting ciphertexts, if they are equal then her and Bob’s inputs are equal.
6. Alice may inform Bob about the result.

This simple protocol has been extended in [8] into a PE protocol that enables two parties Alice and Bob to privately compare their sets of unique values $V_A$ and $V_B$ respectively, and allows the requesting party, Alice, to retrieve some extra information about records in $V_B$ that match records in $V_A$. The PE protocol includes the following steps:

1. Both Alice and Bob apply hash functions $h$ to all the elements in their sets, so that $X_A = h(V_A)$ and $X_B = h(V_B)$. Alice chooses a secret PH key $E_A$ at random, and Bob chooses two PH keys $E_B$ and $E_B'$, all from the same group $Z_p$.
2. Alice encrypts the set: $Y_A = E_A(X_A) = E_A(h(V_A))$
3. Alice sends to Bob her hashed set $Y_A$, reordered lexicographically.
4. Bob encrypts each entry $y \in Y_A$ with both $E_B$ and $E_B'$, and for each $y$ sends back to Alice 3-tuple $(y, E_B(y), E_B'(y))$.
5. For each $h(v) \in X_B$, Bob does the following:
   (a) Encrypts $h(v)$ for use in equality test using $E_B$.
   (b) Encrypts $h(v)$ for use as a key to lock the extra information about record $v$, referred to as $ext(v)$, using the $E_B'$ keys: $v \in y \in Y_A$, reordered lexicographically.
   (c) Encrypts the extra information: $c(v) = K(v, ext(v))$
      Where $K$ is a symmetric encryption function.
   (d) Forms a pair $(E_A(h(v)), c(v))$
      These pairs, containing a private match element and the encrypted extra information about record $v$, are then transferred to Alice.
6. Alice removes her encryption $E_A$ from all entries in the 3-tuples received at Step 4 obtaining tuples $a$, $\beta$, and $\gamma$ such that $(a, \beta, \gamma) = (h(v), E_B(h(v)), E_B'(h(v)))$. Thus, $a$ is the hashed value $v \in V_A$, $\beta$ is the hashed value $\gamma$ encrypted using $E_B$, and $\gamma$ is the hashed value $v$ encrypted using $E_B'$.
7. Alice sets aside all pairs received in Step 5, whose first entry is one of the $\beta$ tuples obtained in Step 6. Then using the $\gamma$ tuples as symmetric keys it decrypts the extra information contained in the second entry in the pair.

Similar results could potentially be achieved when 1-out-of-$n$ PEqT protocol would be combined with 1-out-of-$n$ Oblivious Transfer protocol as demonstrated in [5]. However, as pointed out in Section II mixed solutions usually do not blend well as each requires its own computationally expensive preparation phase.

IV. INVESTIGATORY DATA ACQUISITION PROTOCOL

The application scenario and generalization of the problem statement provided in Section I are the motivation behind creation of the Investigatory Data Acquisition Protocol (IDAP). IDAP combines 3Pass protocol and the PE protocols presented in Section III in order to allow private matching against $i$ different input parameters $ip_{1,i}$.

The PE protocol allows conditional retrieval of extra information $ext(v)$ for records $v$ that are common between chooser and sender. Consequently the PE would be a suitable solution for the problem at hand when $i = 1$, but if there is more than one input $ip$, then the following IDAP protocol can be used to retrieve the records:

1. Chooser provides the sender with names of $k$ return parameters $rp$ and names of $i$ input parameters $ip$.
2. Based on the requirements received in Step 1, the sender gathers all $rp$ and $ip$ parameters for all the records into a common SQL view referred to as the source, i.e., temporary SQL table. The chooser is provided with the schema of this view, describing the data formats used.
3. For each $ip$ the sender creates a list of names, or distinct, values. Each such value is associated, for the duration of the enquiry, with a randomly generated commutative key $E_{E_{\eta(ip)}}$.
4. Sender iterates through the source and does the following for each data row $v$:
   (a) Select a random symmetric encryption key $\kappa(v)$.
   (b) Use $\kappa(v)$ to encrypt list of the $k$ parameters $E_{\eta(ip)}(v)$
   (c) Encrypt $\kappa(v)$ using $i$ commutative keys $E_{E_{\eta(ip)}}$ associated with the values of $ip$ parameters in row $v$.
   (d) Send to chooser pairs $(a, \beta)$ made up from:
      $a$ - encrypted $\kappa(v)$ from Step 4(c),
      $\beta$ - encrypted list of $rp$ from Step 4(b) reordered lexicographically.
5. For each $ip_{e}$ chooser engages in a PE protocol to retrieve random commutative keys associated with the interesting value $x_e$.
6. Chooser attempts to decrypt each $a$ in the pairs received in Step 4(d) using all $k$ commutative keys retrieved in Step 5.
7. In Step 6 chooser obtained key $\kappa(v)$ protecting the return parameters encrypted by sender in step 4(b) and provided to the chooser in Step 4(d). Finally $rp$ parameters for each data row matching the selection criteria $ip_{a} = x_{a} : 1 < \alpha < i$ are obtained.
If a given pair \(\langle \alpha, \beta \rangle\) sent to chooser in Step 4(d) would not match the selection criteria then depending on the commutative encryption scheme used for locking the key \(\alpha(v)\) decryption error would be raised in Step 6 or Step 7. Thus, when PH scheme would be used then no errors would be detected at Stage 6, but the symmetrical decryption process in Step 7 would rise and error, and would not be capable of decrypting the record. In turn if the CEG would be employed then by the use of the checksums it would be discovered in Step 6 that the decryption commutative keys at hand are incapable of decrypting the ciphertext.

Of an interest is the fact that use of CEG in the above protocol would reveal existence of records that match few but not all input parameters. Under CEG the ciphertext is made up of n 4-tuples, where n is the number of times a given value has been encrypted or locked. Each 4-tuple contains a checksum that can be used in conjunction with the decryption key to verify that a given ciphertext has been locked by a corresponding encryption key. Consequently when CEG is used, the chooser will know how many records in the source match the value \(x_i\) for a given input \(ip_o\). Depending on the circumstances this may be beneficial to the enquiry, and acceptable by the data controller.

The above protocol requires a perfect match between the input parameters \(x_{1-i}\) and the respective values in the source. Thus, going back to the application scenario from Section I, the police could request to obtain data on all female employees that are 26 years old and have a white ethnic origin. However, such a request would not return any results for the records where a data subject is 25 or 27 years old. It is impossible to create a fuzzy match based on the technologies employed, but the police could ask the data controllers to create ranges of values acceptable for a given parameter. In such solution all the records where a data subject would be in their teens, twenties, thirties and so on, could be encrypted using keys common for their age range, and the PE in Step 5 would also use this range name or unique identifier to collect the related commutative key.

V. EVALUATION

The communicational complexity of the IDAP is quite small. It is possible to run the protocol with only \(O(2 + 2i)\) data transfers. This property would allow for non-interactive runs of the protocols where the public authorities submit their requests and later return for the results. Consequently, going back to the application scenario businesses that are not yet online could use a simple database of their employees or even a spreadsheet as source and exchange information with the authorities in person or via post.

IDAP is computationally expensive, however, being a protocol designed for a specific task of allowing privacy enhanced investigations to take place. The time taken for the protocol to complete a run is negligible in the scale of an inquiry conducted by the public authorities. Complexity of the protocol is dependent of the underlying data, since each unique value for all input parameters needs to be treated separately. Thus, the computational complexity for cases where the input parameters have only two possible values will be at its minimum, and when all values of \(ip_{1-i}\) are unique it will be at its maximum. Assuming that the row count of the source is \(n\), the computational complexity of the protocol is shown in Table I.

<table>
<thead>
<tr>
<th>Step</th>
<th>Symmetric Crypto</th>
<th>Asymmetric Crypto</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>key generation</td>
<td>crypto. operation</td>
</tr>
<tr>
<td>Step 3</td>
<td>(\min)</td>
<td>(-)</td>
</tr>
<tr>
<td></td>
<td>(\max)</td>
<td>(-)</td>
</tr>
<tr>
<td>Step 4</td>
<td>(\min)</td>
<td>(O(n))</td>
</tr>
<tr>
<td></td>
<td>(\max)</td>
<td>(O(n))</td>
</tr>
<tr>
<td>Step 5</td>
<td>(\min)</td>
<td>(O(n))</td>
</tr>
<tr>
<td></td>
<td>(\max)</td>
<td>(O(n))</td>
</tr>
<tr>
<td>Step 6</td>
<td>(\min)</td>
<td>(-)</td>
</tr>
<tr>
<td></td>
<td>(\max)</td>
<td>(-)</td>
</tr>
<tr>
<td>Step 7</td>
<td>(\min)</td>
<td>(O(n))</td>
</tr>
<tr>
<td></td>
<td>(\max)</td>
<td>(-)</td>
</tr>
<tr>
<td>Total</td>
<td>(\min)</td>
<td>(O(n^2 + i))</td>
</tr>
<tr>
<td></td>
<td>(\max)</td>
<td>(O(n(2 i + 1)))</td>
</tr>
</tbody>
</table>

Cost in ms/operation: 0.66 0.33 10 50

The complexity of each of the steps in the IDAP protocol, providing min. and max. value possible for given \(n\) and \(i\). Where \(n\) is the number of the data rows in the source table or view, and \(i\) is the number of input parameters in the query. Cost in ms for performing given operation from managed C# .NET prototype application is also given.

Table I shows that the total time for the enquiry is linear to both \(i\) – the number of input parameters \(ip\); and \(n\) – number of the data rows in the source table or view. Fig. 3 and 4 illustrate this concept. The maximum time required for an investigation with five input parameters on 1000 rows of data can be read from Fig. 4 as being just over 800s. These figures were calculated from average cost taken to perform the operations under C# .NET managed code on a PC with 1GB of RAM and a 1.8GHz CPU.

![Fig. 3 Protocol run time depending on the number of input parameters.](image-url)
The max. (and min.) time needed for the IDAP enquiry is linear to the number of input parameters \( ip \) used as selection criteria. Here the maximum time required is show by the solid line and the minimum is shown by the dashed line. Constant number of data records \( n = 100 \) was used in this graph.

VI. CONCLUSION

This paper has shown that creation of privacy-preserving IDAP that allows for more than one private selection criteria in a retrieval protocol is possible. The solution shown is based on the well-known secret exchange primitive of 3Pass and the most scalable symmetric PIR protocol found in literature, PE by Agrawal, Evfimievski and Srikant. The complexity of the protocol is relatively high, but taking into consideration that the protocol is designed with real-life investigations, where per case permissions must be granted to retrieve third-party data, time taken per enquiry is negligible. This makes the IDAP protocol valid as a non-intrusive investigation technique for public authorities.

REFERENCES

Minimising Collateral Damage: Privacy-Preserving Investigative Data Acquisition Platform

Zbigniew Kwecka
School of Computing, Edinburgh Napier University, UK

Prof. William J. Buchanan
School of Computing, Edinburgh Napier University, UK

ABSTRACT
Investigators often define invasion of privacy during their operations as collateral damage. Inquiries that require gathering data from third parties, such as banks, Internet Service Providers (ISPs) or employers are likely to impact the relationship between the data subject and the data controller. In this research a novel privacy-preserving approach to mitigate collateral damage during the acquisition process is presented. This approach is based on existing Private Information Retrieval (PIR) protocols, which due to their computational complexity, cannot be employed in an investigative context. Current PIR protocols are often unsuitable for large databases containing more than few thousand records. This paper provides analysis of the investigative data acquisition process and proposes three modifications that can enable existing PIR protocols to perform investigative enquiries on large databases, including communication traffic databases maintained by ISPs. Consequently, the concept of the Investigative Data Acquisition Platform (IDAP) is formalised in this paper. IDAP is an efficient Symmetric PIR (SPIR) protocol optimised for the specific purpose of facilitating public authorities’ enquiries for evidence. It introduces a semi-trusted proxy into the PIR process in order to gain the acceptance of the general public for trap-door based privacy-preserving techniques. In addition, the dilution factor is defined as the level of anonymity required in a given investigation. Defining this factor allows the investigators to restrict the number of records processed, and therefore, minimise the processing time, while maintaining an appropriate level of privacy. Finally, this paper describes the technique required to facilitate the retrieval of records matching multiple selection criteria.

Keywords: Privacy Enhancing Technology; Data Mining; Data Retrieval
INTRODUCTION

*Those who would give up essential Liberty, to purchase a little temporary safety, deserve neither Liberty nor Safety.* (Benjamin Franklin 11 Nov 1755)

Since the September 11, 2001 many western governments have passed laws empowering public authorities with wider rights to gather operational data (Home Office, 2009; Swire & Steinfeld, 2002; Young, Kathleen, Joshua, & Meredith, 2006). For many years public opinion accepted the invasion of personal privacy rights as the sacrifice needed to fight the terror (Rasmussen Reports, 2008). However, slowly, public opinion is shifting back to a state where such measures are considered unacceptable. This is shown by public opinion surveys, such as the one conducted by Washington Post in 2006 (Balz & Deane, 2006), where 32% of respondents agreed that they would prefer the federal government to ensure that privacy rights are respected rather than to investigate possible terrorism threats. This was an 11% increase from the similar survey conducted in 2003.

In the UK, the public authorities, including Police, request investigative data from third-parties on regular basis (Information Commissioner, 2008) and the data protection legislation allows for such requests, even without warrants (European Parliament, 1995; Home Office, 2007). Depending on the way these requests are performed, human and natural rights of the data-subject can be breached, and/or the investigation can be jeopardized (Kwecka, Buchanan, Spiers, & Saliou, 2008). A recent proposal by the UK government went further and recommended allowing the public authorities direct access to data held by Content Service Providers (CSPs), such as mobile telephony providers and Internet Service Providers (ISPs) (Home Office, 2009).

According to the public consultation document, there were a few major motivating factors behind this proposal, these included: increasing access speeds to records; allowing for covert enquiries by anti-terror and national security agencies; lowering collateral damage to potential suspects under investigation; and enabling the analysis of data to facilitate the profiling of terrorists activities. In response, concerns were raised that if the proposal was implemented, it would thwart the privacy of Internet users around the globe, in order to increase the security of one nation. This research shows that most of the objectives set out in the proposal can still be achieved while maintaining high level of privacy. It is shown that an investigative system can maintain the privacy of the data subjects and also preserve the confidentiality of investigations. However, both security and privacy must be built into the system at the design stage in order to achieve this (Swire and Steinfeld, 2002).

This paper gives an insight into use of Privacy Enhancing Technologies (PETs) in improving the current investigative data acquisition practices. The structure of this manuscript closely follows the methodology used to draw the final conclusions. Section II provides a background to investigative data acquisition and to various privacy-preserving approaches to information retrieval. The analysis of the related research presented in Section II identifies an existing protocol, Private Equi-join (PE) that can facilitate efficient private database searching and information retrieval. It is also shown that the complexity of this protocol is lower than complexity of similar approaches, and for these reason the PE protocol is chosen as the base for
the investigative data acquisition solution. Section III describes this protocol and other commonly used privacy-preserving primitives that can be reused in order to build a platform suitable for investigative enquiries. Section IV provides discussion of the design considerations. The PE protocol is evaluated against the requirements derived from the literature described in Section II. Finally, Section V describes the novelty of this paper – which are the three improvements needed to form an Investigative Data Acquisition Platform (IDAP) based on the PE protocol and the evaluation of IDAP is provided in Section VI. IDAP is an efficient approach to maintain the secrecy, preserving the suspect’s privacy and gaining the public’s support for the PET technologies in digitalised investigative enquiries. The improvements include the introduction of the dilution factor, which is a numeric value that specifies the level of anonymity required for an investigation. Resultantly, the identity of a potential suspect can be hidden in a group smaller than the population, which permits the use of PIR systems with large databases. The second improvement is the technique for building complex privacy-preserving queries without affecting the complexity of the protocol. Finally, a semi-trusted proxy is introduced to the PE protocol, in order to ensure information theoretic privacy of data-subjects that are not the potential suspects.

BACKGROUND AND RELATED WORK
The background section is split into two parts. First the nature of public authorities’ investigations is discussed together with the requirements for the way enquiries for data need to be handled. Then the second part of this section provides background to the private information retrieval protocols.

Data Acquisition
The public authorities are often required to carry out investigations based on data supplied by third parties. Such investigations may include fraud enquiries from Her Majesty Revenue and Customs (HMRC) tax office, solving a crime by Police, investigating alleged terrorism cases by Scotland Yard, or gathering health information about a patient at an Accident and Emergency (A&E) department (Home Office, 2009). The process of obtaining third party records is usually referred to as data acquisition. In the UK there are two major data acquisition legislations available to the public authorities, these are: the Data Protection Act 1998 (DPA) and The Regulation of Investigatory Powers Act 2000 (RIPA), but similar legislations can be found across Europe. Depending on the nature of the investigation, and the type of data required, the public authorities choose between the above legislations as grounds for their data acquisition requests. The main difference is that the voluntary disclosure mechanism of DPA can be used to gather general data records, while RIPA regulates requests for communication data. For any requests made under the DPA investigators need to provide justification for the acquisition requests and the dataholder can refuse providing data with no legal consequences if the request is not accompanied by a legal warrant (Spiers, 2009). Data acquisition notices served under RIPA do not need any form of justification and the dataholder will face a penalty if the relevant data is not provided to requesting public authority within two weeks. Still, the dataholder may choose to accept the penalty and refuse to provide the communication data (Home Office, 2007). Since, the searches of the communication data can be labour intensive, depending on the information
system employed by a given CSP, the public authorities must make a contribution towards the costs incurred by the dataholder during fulfilling the data acquisition notice served under RIPA. Evidence handling guidelines suggest that any evidence collected may need to be presented in front of a court of law. If such requirement arises the electronic evidence must be provided as a true image of the data gathered (Association of Chief Police Officers, 2003). For these reasons data acquired from third parties should be handled as potential evidence: no action taken should alter the data; and full audit trail needs to be kept for the processing of the data. Digital forensic literature suggests that in some cases there can be a number of suspects in a forensic inquiry and the enquiries may become complex (Palmer, 2001), thus a system for investigative data acquisition would need to accommodate for request with multiple interesting records and multiple selection criteria.

Retrieval of Private Information
Leaving the investigative context aside, the retrieval of information from a third-party in a private manner is a generic problem that has been researched for use in a variety of different scenarios. In the first instance, Private Information Retrieval (PIR) protocols were designed with a basic requirement of acquiring an interesting data record, from a dataholder (the sender) in a way that stops this dataholder from identifying which record is of interest to the requestor (the chooser). These protocols were not concerned with the secrecy of the records stored in the database, thus, in its least optimised state, a PIR could have been achieved by transferring the whole database from the sender to the chooser, as this would allow the chooser to retrieve the record in a private manner. Consequently, the main motivation behind PIR schemes is the achievement of minimal communicational and computational complexity (Ostrovsky & William E. Skeith III, 2007). A stronger notion than PIR is the 1-out-of-n Oblivious Transfer (OT) primitive that allows the retrieval of a randomly-selected record from a dataset of n elements held by the sender, in a way that the sender cannot learn which record has been transferred, and the chooser cannot learn anything about other records in the dataset (Schneier, 1995). 1-out-of-n OT protocols, that allow chooser to actively select a record to be retrieved, and that have linear or sub-linear complexity, can be referred to as Symmetric PIR (SPIR) protocols, as they protect the records of both parties during the information retrieval. These useful privacy-preserving data retrieval protocols can be employed in a variety of systems: electronic watch-lists of suspects (Frikken & Atallah, 2003); cooperative scientific computation (Du & Atallah, 2001; Goldwasser & Lindell, 2002); and on-line auctions (Cachin, 1999).

With the use of SPIR protocols a chooser would be capable of privately retrieving a record from the sender’s database, by secretly referring to its index in this database. Such an index is expected to be publically available in an electronic catalogue or a directory (Aiello, Ishai, & Reingold, 2001; Bao & Deng, 2001). However, organisations, with large private databases, cannot be expected to maintain such freely available indexes. Also, it is expected that an investigator would normally refer to a suspect by a name, ID or phone number. For this reason, before the data can be received using SPIR, a search would need to be performed by the chooser against the records in the sender’s database. Such a private search operation requires a protocol that allows two parties to compare their values in a private manner. The protocols that are optimised to make comparisons for equality are referred to as Private Equality Test (PEqT)
protocols, which are often based on commutative (Frikken & Atallah, 2003; Kwecka et. al. 2008), or homomorphic, cryptosystems (Bao & Deng, 2001). An interesting record can thus be located in a database using a 1-out-of-n PEqT protocol and then retrieved with help of SPIR. Often each of these protocols would have a separate computationally expensive preparation phases, and such a solution would not be optimal. The exception to this rule is a range of protocols including: private intersection; private intersection size; and the PE defined in (Agrawal, Evfimievski, & Srikant, 2003). These protocols are based on commutative encryption, and thanks to the use of different properties of the underlying commutative algorithms, are capable of allowing for both private matching and private data retrieval. The operation of the PE is described in Section III together with a brief introduction of the cryptographic mechanisms used by this protocol, and other commonly used privacy-preserving primitives.

BUILDING BLOCKS
This section describes PE protocol that has been identified as suitable to facilitate private data acquisition enquiries. It relies on commutative cryptography, and thus some background for this is provided.

Commutative Cryptosystems
Many cryptographic applications employ sequential encryption and decryption operations under one or more underlying cryptosystems. The reasons to sequence (cascade) different cryptographic schemes together include: strengthening the resulting ciphertext; and achieving additional functionality, which is impossible under any given encryption scheme on its own (Shannon, 1949; Weis, 2006). A basic cascadable cryptosystem can consist of a number of encryption stages, where the output from one stage is treated as the input to another. In such a basic cascadable cryptosystem it is necessary to decrypt in the reverse order of encryption operations. However, a special class of sequential cryptosystems – commutative cryptosystems – allows for the decryption of a ciphertext in an arbitrary order. Thus, a ciphertext \( c = e_a e_b(m) \) (\( c \) – ciphertext, \( m \) – plaintext, \( e \) – encryption operation under keys \( a \) and \( b \)), could be decrypted as either \( m = d_b d_a(c) \) or as \( m = d_a d_b(c) \). The advantages of such cryptosystems were widely promoted by Shamir (1980) and used in his, Rivest’s and Aldman’s, now classic, game of mental poker, employing the Three-Pass (3Pass) secret exchange protocol.

PH encryption algorithm
The most commonly used commutative cryptosystem is based on the Pohling-Hellman (PH), asymmetric private key scheme (Pohling and Hellman, 1978). While the PH protocol influenced the design of the ever-popular Rivest-Shamir-Adleman (RSA) public key scheme (Rivest, Shamir and Adleman, 1978), it has never become popular since it is asymmetric, and therefore, slow in comparison to other private key systems. The main strength of PH is that it is commutative for keys based on the same prime number, and that it allows for comparing the encrypted ciphertexts. Consequently, under PH the two ciphertexts \( c_{ab} = e_a e_b(m) \) and \( c_{ba} = e_b e_a(m) \) hiding the same plaintext \( m \) are equal (1), while this is not the case with ordinary encryption protocols, in which condition (2) is satisfied. The RSA shares these properties, however, it is
unpractical and unsafe to share the composite modulus of RSA between the communicating parties.

\[ e_1e_2(m) = e_1e_2(m) \]  \hspace{1cm} (1)

\[ e_1e_2(m) \neq e_1e_2(m) \]  \hspace{1cm} (2)

Thanks to these properties PH can be used in the 3Pass primitive that allows two parties to exchange data without exchange of keys, as well as to perform PEqT that permits private matching of data records.

Since, the popular public-key algorithm RSA is based on the PH scheme they are similar in operation, where the PH encryption and the decryption functions are shown in (3) and (4) respectively. In both PH and RSA the cryptographic operations are based on modular exponentiation, and different exponents (keys) are used for encryption (exponent \( e \)) and decryption (exponent \( d \)). However, in case of PH the operations are performed modulo a large prime number \( p \), while the RSA uses a modulus made out of two prime numbers.

\[ c = m^e \mod p \]  \hspace{1cm} (3)

\[ m = c^d \mod p \]  \hspace{1cm} (4)

The encryption exponent \( e \) is chosen randomly from the range (5) and then it is used to calculate the decryption exponent \( d \) (6):

\[ 1 < e < p - 1 \]  \hspace{1cm} (5)

\[ de \equiv 1 \mod(p - 1) \iff d = e^{-1} \mod(p - 1) \]  \hspace{1cm} (6)

Unlike it is the case with RSA in PH it is easy to calculate the decryption key knowing the modulus (prime \( p \)) and the encryption key, thus both keys should be kept secret. Hence, this is private key protocol. However, there is no harm in making the large prime \( p \) public. An adversary with the knowledge of the ciphertext \( c \) and the prime \( p \) would need to solve the following hard problem to break the PH protocol (Schneier, 1995):

\[ e = \log_p C \mod p \]  \hspace{1cm} (7)

It is also worth noting that both RSA and PH may leak some information about the plain text, i.e. its parity, and for these reason the inputs to these cryptographic operations should be almost random, such as secret encryption keys and hashed signatures.

**Three Pass Protocol**

The 3Pass protocol, shown in Fig. 1, was intended to allow two parties to share a secret without exchanging any private or public key. The protocol was aimed at providing an alternative to public-key encryption and DH-like key negotiation protocols.
The operation of the protocol can be described using the following physical analogy:

1. Alice places a secret message $m$ in a box and locks it with a padlock $E_A$.
2. The box is sent to Bob, who adds his padlock $E_B$ to the latch, and sends the box back.
3. Alice removes her padlock and passes the box back to Bob.
4. Bob removes his padlock, and this enables him to read the message from inside the box.

There could be more parties, or encryption stages, involved in a 3Pass-like protocol, and this property makes it ideal for locking a plaintext message multiple times and then unlocking it in an arbitrary order.

**Private Equality Test**

PE$qT$ protocols can be used to privately verify whether two secret inputs are equal or not. Agrawal, Evfimievski and Srikant (2003) proposed one of the most scalable and flexible PE$qT$ protocols for operations on datasets. The scheme is illustrated in Fig. 2 and can be described in the following steps:

1. Alice encrypts her input and sends it to Bob.
2. Bob encrypts the ciphertext received from Alice and sends it back.
3. Bob encrypts his secret input and sends it to Alice.
4. Alice encrypts the ciphertext containing Bob’s input.
5. Alice compares the two resulting ciphertexts, if they are equal then her and Bob’s inputs are equal.
6. Alice may inform Bob about the result.
Private Equality Test allows two parties to test their inputs for equality, without revealing these inputs.

**Private Equi-join Protocol**

PE protocol can enable two parties, the chooser and the sender, to privately compare their sets of unique values $V_C$ and $V_S$, and allows the chooser to retrieve some extra information $\text{ext}(v)$ about records $V_S$, that match records $V_C$ on a given parameter. The PE protocol involves the following steps:

1. Both parties apply hash function $h$ to the elements in their sets, so that $X_C = h(V_C)$ and $X_S = h(V_S)$. *Chooser* picks a secret PH key $E_C$ at random, and *sender* picks two PH keys $E_S$ and $E_S'$, all from the same group $\mathbb{Z}_p$.
2. *Chooser* encrypts entries in the set: $Y_C = E_C(X_C) = E_C(h(V_C))$.
3. *Chooser* sends to *sender* set $Y_C$, reordered lexicographically.
4. *Sender* encrypts each entry $Y_S \in Y_C$, received from the *chooser*, with both $E_S$ and $E_S'$ and for each returns 3-tuple $(y, E_S(y), E_S'(y))$.
5. For each $h(v) \in X_S$, *sender* does the following:
   (a) Encrypts $h(v)$ with $E_S$ for use in equality test.
   (b) Encrypts $h(v)$ with $E_S'$ for use as a key to lock the extra information about $v$, $\kappa(v) = E_S'(h(v))$.
   (c) Encrypts the extra information $\text{ext}(v)$:
      $$ c(v) = K(\kappa(v), \text{ext}(v)) $$
      Where $K$ is a symmetric encryption function and $\kappa(v)$ is the key crafted in Stage 5b.
   (d) Forms a pair $(E_S(h(v)), c(v))$. These pairs, containing a private match element and the encrypted extra information about record $v$, are then transferred to *chooser*.
6. *Chooser* removes her encryption $E_C$ from all entries in the 3-tuples received in Step 4 obtaining tuples $\alpha$, $\beta$, and $\gamma$ such that $(\alpha, \beta, \gamma) = (h(v), E_S(h(v)), E_S'(h(v)))$. Thus, $\alpha$ is the
hashed value $v \in V_c$, $\beta$ is the hashed value $v$ encrypted using $E_s$, and $\gamma$ is the hashed value $v$ encrypted using $E'_s$.

7. **Chooser** sets aside all pairs received in Step 5, whose first entry is equal to one of the $\beta$ tuples obtained in Step 6. Then using the $\gamma$ tuples as symmetric keys it decrypts the extra information contained in the second entry in the pair $(E_s(h(v)), c(v))$.

The above protocol can perform the basic functions required for the purpose of investigative data acquisition. Its use in investigative scenarios is described in Section IV.

**DESIGN**

This section derives the requirements for an investigative data acquisition process from the literature discussed in Section II. The PE protocol has been initially selected as the most suitable base for the acquisition platform, thus, it is evaluated against these requirements and its shortcomings are identified.

According to the literature discussed in Section II, the protocol chosen for data gathering should allow for the retrieval of a number of interesting records at the time. If this is not the case multiple sequential runs of the protocol should bear low computational and communicational overhead. The protocol must leave the dataholder in control of the data, since the data retrieval can only be performed with the dataholder’s consent. Taking into consideration that a dataholder has two weeks to provide the data under RIPA the computational complexity of the protocol can be reasonably large. However, shortening the time required by the data acquisition process is one of the main reasons that the government provide as a justification to the proposed modernisation of the process. Data records should be retrieved from the dataholder on a record-by-record basis, so that if only one of many records is required for the investigation, other records can be discarded. Otherwise the public authorities can end up storing large amount of unnecessary data, and this can prove costly taking into consideration the level of security and auditing involved. Finally, the cost of the solution should be low as the public authorities will have to cover the costs of running the system. If the costs were not covered by the authorities, the dataholders would transfer the costs of handling the enquiries to the end-users and such a solution would be unacceptable.

A query, such as data acquisition request, made against a modern relational database can be mapped using the Structured Query Language (SQL). Thus, the operations required could be split into the following:

1. **Identification of the type of the information that is required.** These could be $h$ parameters that contain answers to investigator’s questions, referred to as return parameters $rp_{j,h}$, for example DoB, address, location of a card payment, or the telephone numbers called by a given subscriber.
2. **Specification of any circumstantial request constraints, or $l$ different input parameters, $ip_{j,l}$, with values $ip_{val_{j,l}}$, such as a time frame of the transactions being requested.**
3. **Specification of the relevant data subject, such as by identifying the individual whose data is to be retrieved, or by providing the mobile phone number of the suspect, and so on.** This parameter is referred to as the record of the interest $ri$, with value of $ri_val$. 
Then, if we refer to the dataset as the source, the request for investigative data could be mapped into the following SQL query:

\[
\text{SELECT } r_{p1}, r_{p2}, \ldots, r_{ph} \\
\text{FROM source} \\
\text{WHERE } r_{i} = r_{i\_val} \text{ AND } i_{p1}=i_{p\_val1} \text{ AND } i_{p2}=i_{p\_val2} \text{ AND } \ldots \text{ AND } i_{pl}=i_{p\_vall} \tag{8}
\]

In most cases the names of the return parameters, as well as the names of the input parameters, and values of these input parameters, can be openly communicated. But the value of the interesting record, \( r_{i\_val} \) is used to uniquely identify the suspect and must be hidden. This can be achieved by running a database query for the return parameters of all the records that satisfy the conditions defined by the input parameters, and then collecting the interesting record from the sender using the PE protocol. Consequently, the query that is actually run on the sender’s database can be rewritten as:

\[
\text{SELECT } r_{i}, r_{p1}, r_{p2}, \ldots, r_{ph} \\
\text{FROM source} \\
\text{WHERE } i_{p1}=i_{p\_val1} \text{ AND } i_{p2}=i_{p\_val2} \text{ AND } \ldots \text{ AND } i_{pl}=i_{p\_vall} \tag{9}
\]

The results of query (9) would then be an input to the PE that would enable the chooser to privately select only the record of interest that matches given \( r_{i\_val} \).

**Base Protocol**

Section II discussed different types of protocols available that could enable the chooser to download a record from the sender’s database maintaining the secrecy of the record selected. It also mentioned that most available protocols could not achieve the searches necessary for the acquisition protocol, and that a combination of two or more protocols is required. Such a combination typically results in high computational and communicational complexity, because each protocol usually requires its own preparation phase. The PE protocol was identified as the most suitable to become the base protocol for the data acquisition, since it can be used to for both private matching and SPIR, thus, has a low computational overhead. Table I illustrates the computational complexity of this protocol.

There are also other factors that make PE an ideal base protocol for the task at hand. It allows for acquiring more than one interesting record at the time, and adding more records to the enquiry increases the processing by negligible five asymmetric key operations and one symmetric operation per each extra interesting record in an enquiry. Use of the PE allows the dataholder to remain in full control of data, and to decide which data can be disclosed. In the PE protocol each record is processed separately and there are no chances of the records being mixed up by the privacy-preserving process. Thanks to this fact unnecessary, data of non-suspects could be discarded on reception by the authorities and still the encrypted interesting records received would form valid evidence for use in a court of law. The costs involved in building and
deploying PE based solution to the private data acquisition are anticipated to be low since it is a software system and the architecture would be based on the protocol from the public domain.

TABLE I

<table>
<thead>
<tr>
<th></th>
<th>Symmetric Crypto.</th>
<th>Asymmetric Crypto.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>crypto. operation</td>
<td>key generation</td>
</tr>
<tr>
<td>Step 1</td>
<td>-</td>
<td>$O(3)$</td>
</tr>
<tr>
<td>Step 2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 4</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 5</td>
<td>$O(n)$</td>
<td>-</td>
</tr>
<tr>
<td>Step 6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 7</td>
<td>$O(m)$</td>
<td>-</td>
</tr>
<tr>
<td>Total Complexity</td>
<td>$O(n + m)$</td>
<td>$O(3)$</td>
</tr>
</tbody>
</table>

The complexity of different steps of the PE protocol. Where $n$ is the number of the data rows in the source, and $m$ is the number of interesting records.

The processing time required for the protocol to run is the main drawback of the PE protocol. If there are thousands of records in the database it only takes few minutes for the complete run of the protocol, however, the processing time is linear to the number of records in a dataset and data retrieval from a database with few million records would take few days to run on a typical PC. During an urgent enquiry, such as when the life of an individual is in danger, or where an individual can seriously endanger others, police should be able to get access to data within minutes. Such result could not be expected from the PE protocol if the database has more than a few thousand records. Additionally, even if the data requested is relatively small in size, for example 100kB per record, the results from a database of five million records would require more than 500MB of data to be transferred over the Internet. Clearly, there is a requirement for the PE to return a subset of the sender’s database, rather than the whole database or another solution would need to be chosen.

Another issue is that the PE-based system allows for secure matching on a single value per record, for example an IP address, suspect’s name, or a credit card number. In some scenarios it may be required to request records based on a number of secret input parameters. Consider scenario where Police has a profile of a suspect (e.g. sex, age, and ethnic origin) and would like to find individuals fitting this profile working in organizations in a neighbourhood to the crime scene, but revealing the profile to these organizations may harm the investigation and the individuals matching the profile. Currently the police would often have to delay their enquiries in order to protect the investigation, and the innocent individuals fitting the profile. For example, if the case being investigated had a public tension around it, and the suspect’s profile matched individuals in a local minority group, an openly conducted enquiry could have serious consequences to the members of this minority.

Finally, the idea of a system that requires the legitimate dataholders to transfer data about all the records in the database to the third-party for investigative purposes is controversial. In PE the
investigators can only read the data requested, still the records of all data subjects would require processing. This would not be normally allowed under the DPA. Also, it is likely that the general public would be concerned about trap-door, polynomial time secure, cryptographic technique protecting personal data. Consequently, there is a requirement for a control that would limit the possibility of exposure of the private records unrelated to the investigation.

**NOVEL APPROACH TO DATA ACQUISITION - IDAP**

The previous section has listed the design considerations for using PE as the base protocol for data acquisition in the investigative context. This section addresses the shortcomings of PE by three different correcting measures that modify the PE protocol into IDAP.

**Allowing multiple selection criteria**

The PE protocol can be used to privately retrieve data if the data is identified by a single parameter, such as ID number, credit card number, IP address, and so on. However, this is not always the case. Consequently, if the protocol needs to be used to find a suspect based on circumstantial knowledge, or a suspect’s profile, the PE protocol would need to be modified. Query (10) shows the way the request (9) would be modified for such enquiry, here \( sip_{1:j} \) stands for \( j \) secret input parameters:

```
SELECT sip_1, sip_2, ..., sip_j, rp_1, rp_2, ..., rp_h
FROM source
WHERE ip_1 = ip_val_1 AND ip_2 = ip_val_2 AND ... AND ip_l = ip_val_l
```

A computationally expensive solution to this problem has been published by Kwecka, Buchanan, and Spiers (2010), who suggest that the symmetric encryption should be used to lock the return parameters and the symmetric keys should be secured with relevant commutative encryption keys that are unique to each value of the secret input parameter returned for the given row. Despite being computationally expensive, their solution has a unique benefit of allowing semi-fuzzy matching of the results if the underlying commutative protocol is ElGamal-based. In this paper a simplified approach is proposed. Since, query (10) replaces the \( ri \) parameter with \( j \) different \( sip \) parameters, the list of these \( j \) parameters could be used as a complex \( ri \) in the improved IDAP protocol. Thus, in Steps B_2 and A_1 a list of all values for the \( sip \) parameters would be hashed together. In this way neither the security of the PE protocol, nor its complexity, are affected by this improvement.

**Lowering Processing Time**

Section IV recommended minimising the processing time required for each run of the protocol in large databases, such as those belonging to ISPs and mobile telephony providers. Theoretically, in order to maintain the privacy of the suspect, the chooser needs to request all the records in the database to be included in a given run of the PE protocol. This is the only way that no information about interesting record is revealed and the correctness of this scheme can be proven. In its current form, the PE protocol would not be capable of processing any urgent requests due to the processing overhead. The mitigation for this could be to limit the numbers of
records processed per enquiry by the sender. Privacy of the alleged suspect should be protected, but, if the probability of the sender guessing the ID of the interesting record is, for example 1:1000, and not 1:n, and the dataholder has no other information that could help infer any knowledge as to the identity of the suspect, then this research argues that the privacy of the suspect and the investigation is maintained. Police sources suggest that occasionally during traditional data gathering enquiries, this is face-to-face, investigators would use a concept of diffusion - hiding the suspect’s identity by asking open-ended questions about a larger group of individuals rather than about a single person. This is a widely-accepted technique, however, in the digitalised environment, a system that would maintain privacy while providing answers to such general questions does no currently exist. Consequently, any attempts of investigators to cast their net wide during electronic investigations are prohibited and treated as fishing for evidence.

The problem is to decide on the technique of narrowing down the scope in a way that ensures interesting records are among the results returned. If the list of the record identifiers is public, such as the list of the Internet Protocol (IP) addresses or telephone numbers served by a given network operator, then the chooser could simply selected records to be processed at random from such a list. However, if such a list is not available, it would be possible to split PE protocol into separate parts: PEqT; OT; and an additional off-line preparation phase. In this way the initial off-line phase could be run against the whole database, but the information retrieval would be performed against a smaller set of records. Thus, the number of records requested per each interesting record can be defined as the diluting factor – o, and the modified PE protocol would operate as follows:

Phase A - Preparation
1. **Sender** applies hash function $h$ to the elements in the input set $V_S$, so that $X_s = h(V_s)$.
2. **Sender** picks an encryption PH key $E_s$ at random from a group $Z^*_p$, where $p$ is a strong prime.
3. **Sender** encrypts each $h(v) \in X_s$ with the key $E_s$, the result is a list of encrypted identities $Y_s = E_s(h(V_s))$.

If more record need to be added to the set these can be processed using Steps A1 (Step 1 of Phase A) and A3, and then added to the list.

Phase B - PEqT
1. Following a request for data, **sender** provides **chooser** with a complete list of encrypted identities prepared during Phase A, reordered lexicographically.
2. **Chooser** applies hash function $h$ to the elements in set containing the identities of the interesting records, so that $X_c = h(V_c)$.
3. **Chooser** picks a commutative cryptography key pair, encryption key $E_c$ and decryption key $D_c$, at random from the same group $Z^*_p$ that was used by **sender** in the Phase A.
4. **Chooser** encrypts entries in the set $X_c$, so that: $Y_c = E_c(h(V_c))$.
5. **Chooser** sends to **sender** set $Y_c$, reordered lexicographically.
6. **Sender** encrypts with key $E_s$ each entry $y \in Y_c$ received from **chooser**.
7. **Sender** returns a set of pairs \( \{y, E_s(y)\} \) to **chooser**.

8. **Chooser** decrypts each entry in \( E_s(Y_c) \), obtaining \( E_s(X_c) = D_c E_s(E_c(X_c)) = D_c E_s(Y_c) \).

9. **Chooser** compares each entry in \( E_s(X_c) \) to the entries of \( Y_s \) received in the Step B1. In this way the interesting records can be identified.

**Phase C - OT**

1. After identifying the interesting records in \( Y_s \) the **chooser** selects at random \( o-1 \) other unique records from \( Y_s \) for each interesting record in \( V_c \). These are the diluting records, that together with the records of interest, form a shortlist for the enquiry. If the number of interesting records multiplied by \( o \) is greater than \( n \), the size of the dataset \( V_s \), then the complete \( Y_s \) is shortlisted.

2. Send the shortlist to **sender**.

3. **Sender** picks an encryption PH key \( E_s' \) at random from the group \( Z_n^* \).

4. **Sender** identifies entries \( h(v) \) from \( X_s \) that have been shortlisted and processes each shortlisted record in the following way:
   
   (a) Encrypts \( h(v) \) with \( E_s' \) to form the key used to lock the extra information about \( v \), i.e. \( \kappa(v) = E_s'(h(v)) \).

   (b) Encrypts the extra information using a symmetric encryption function \( K \) and the key \( \kappa(v) \) crafted in the previous step:

   \[
   c(v) = K(\kappa(v), \text{ext}(v))
   \]

   (c) Forms a pair \( \{E_s(h(v)), c(v)\} \).

5. The pairs formed in C4(c), containing a private match element and the encrypted extra information about record \( v \), are then transferred to **chooser**.

6. **Sender** encrypts each entry \( y \in Y_c \), received from **chooser** in Step B3, with key \( E_s' \) to form set of pairs \( \{y, E_s'(y)\} \).

7. Pairs \( \{y, E_s'(y)\} \) are then transferred to **chooser**.

8. **Chooser** removes the encryption \( E_c \) from all entries in the 2-tuples received in Step C7 obtaining tuples \( \alpha, \beta \) such that \( \{\alpha, \beta\} = \{h(v), E_s'(h(v))\} \). Thus, \( \alpha \) is the hashed value \( v \in V_c \), and \( \beta \) is the hashed value \( v \) encrypted using \( E_s' \).

9. **Chooser** sets aside all pairs received in Step C5, whose first entry is equal to one of the first entry of any two-tuples obtained in Step B5. It then uses the appropriate \( \beta \) tuple associated with a given interesting record as a symmetric key to decrypt the extra information contained in the second entry in the pair received in C5. This is performed for all the matching entries.

Fig. 3 illustrates the processes involved in this improved version of acquisition protocol. It is worth noting that there is only five communication rounds required in this protocol. This is two rounds more than in the original PE protocol. Still, most of efficient SPIR protocols require considerably more rounds.
Fig. 3 IDAP Process Flow

Graphical representation of the improved PE protocol
Reassuring the Public
Legal opinion about legality of the protocol that transfers large chunks of non-suspect data to the investigators is divided. Some consider this solution as acceptable as long as it can be proven that the public authorities are unable to decrypt any unsolicited data, while others suggest that anything that creates a privacy risk, however remote, requires the consent of the parties involved. Case law supports both of these opinions, thus, until such case is brought in front of court the matter cannot be answered. Clearly there is a need for a process that would further eliminate the privacy risk to the data records of non-suspects.

Sad quis custodiet ipsos custodies?
But who will watch the watchers? (Juvenal, Satires VI, 347)

It is likely that providing government agencies with encrypted records of innocent, non-suspected individuals would worry the general public. This is despite the data being encrypted in the way that would render the records unusable to the authorities. However, the public may worry that the government organisations have enough computing power to break the encryption used by PE. The solution proposed in this paper in order to reassure the public is to introduce a semi-trusted party into the protocol. This party would be a proxy between the investigators and the dataholder. The following modifications to the PE protocol are proposed:

1. All communication between chooser and sender goes through proxy.
2. Chooser provides proxy with the identifiers of the interesting records encrypted by sender, \( E_h(v) \). This is done over a secure channel, or with the use of the 3Pass protocol, once the parties are authenticated.
3. At the stage where data is transferred from sender in Step C4 (Fig. 3), proxy filters the response and discards the records that were not specified by chooser’s request, that is the records other than the ones identified in the step above.

The semi-trusted party should have no interest in finding out the objective of the investigation, or the content of the data records returned by the dataholder, for this reason it is suggested that the role of this party should be conducted by Information Commissioner’s Office (ICO) or its equivalent in other countries. The party that is chosen must not cooperate with the sender, or the protocol will be broken, since simple matching exercise would reveal the identities of the suspects.

**EVALUATION**

Section V introduced the technique for running queries with multiple selection criteria without affecting the computational complexity of the protocol. As a result the processing time is constant for enquiries with different number of selection criteria, if the time required for the database query is ignored. This is a large enhancement in comparison to the system presented by Kwecka et. al. (2010) where the processing time increases linearly when adding more selection criteria.
Additionally, in the improved protocol the initial processing is dependant on the size of the dataset - \( n \), but it needs to be performed only once in a given period of time, for example once per month, or once per year. The remaining operations are less processing savvy, as illustrated in Table II.

**TABLE II**  
Computational Complexity of IDAP

<table>
<thead>
<tr>
<th></th>
<th>Symmetric Crypto</th>
<th>Asymmetric Crypto</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>crypto. operation</td>
<td>key generation</td>
</tr>
<tr>
<td>Phase A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(run periodically)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 2</td>
<td>-</td>
<td>( O(1) )</td>
</tr>
<tr>
<td>Step 3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Phase B</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(run per enquiry)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>-</td>
<td>( O(1) )</td>
</tr>
<tr>
<td>Step 4</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 8</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Phase C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(run per enquiry)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>-</td>
<td>( O(1) )</td>
</tr>
<tr>
<td>Step 4(a)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 4(b)</td>
<td>( O(m \times o) )</td>
<td>-</td>
</tr>
<tr>
<td>Step 6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 8</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Step 9</td>
<td>( O(m) )</td>
<td>-</td>
</tr>
<tr>
<td>Total Complexity for ( k ) enquiries, where ( n &lt; m \times o )</td>
<td>( O(km(o+1)) )</td>
<td>( O(2k+1) )</td>
</tr>
<tr>
<td>Cost (ms/operation)</td>
<td>0.33</td>
<td>7</td>
</tr>
</tbody>
</table>

The complexity of the different steps in the proposed improved solution. Where \( n \) is the number of the data rows in the source, \( m \) is the number of interesting records. Also the diluting factor \( o \), as well as the number of the protocol runs \( k \) affects the processing time required by the protocol. Cost is the measured average time in ms to perform given cryptographic operation from managed C# .NET code on a Personal Computer with 1.6GHz CPU, 1GB of RAM, running Windows XP.

The proposed method provides significant improvements to the processing time required for enquiries, if total number of records in the *sender’s* database is higher than \( o \times m \), that is higher than the number of interesting records \( m \) multiplied by the diluting factor \( o \). This is illustrated in Fig. 4. However, the true strength of this improvement is seen when multiple enquiries are run on the same database using a single encrypted catalogue of the records, compiled by the *sender* in Phase 1 (shown in Fig. 5).
This proposed modification of the protocol improves significantly the processing time required for the protocol to run for the cases where the product of the number of the interesting records $m$ and diluting factor $o$ is smaller than the number of the records in the database $n$. The graph drawn for $n=1000$ and $o=100$. 

This proposed modification improves significantly the processing time required for the protocol to run for the cases where more than one query is run against the same database. The graph drawn for $n=1000$, $o=100$, and $m=10$. 
The introduction of the diluting factor $o$ will allow real-life enquiries on large databases to be processed in minutes, rather than days as would be the case if off-the-shelf SPIR solution would be employed in the data acquisition process.

In order for the IDAP to be introduced as the platform for investigative data gathering will first need to gain trust of the general public. In an ordinary SPIR protocol all the records are transferred from the sender back to the chooser. However, in IDAP thanks for introducing the concept of dilution the probability that investigators retrieve encrypted records of a particular individual that is not a suspect are small. Thus, for a dataset with $n$ records, during investigation with $m$ interesting records, and the diluting factor $o$, the probability of such event $A$ can be defined as (11).

$$P(A) = \frac{(o-1)m}{n-m}$$

Consequently, for investigation with five interesting records, with a diluting factor of a thousand and dataset consisting a million records, the probability of this event occurring during a single run of the protocol would be less than 0.5%. This also means that the investigators would need to first break the encryption key used by the sender to hide identities (Phase A), before they could attempt to obtain the data about a specific individual that is not a suspect, otherwise the probability of the encrypted data being provided to them would be small. Additionally, if the identity of a data subject is never encrypted under the same key as the data records, investigators would need to successfully brute force two separate keys in order to make use of the retrieved encrypted records. Otherwise the information would be unintelligible.

The merits of the above discussion can improve the perception of the system, however, a key mechanism in IDAP is the semi-trusted proxy. Since, it has no incentives to find out the detail of the investigation, thus it is not going to purchase expensive cutting edge decryption technology to decode the data, nor it is going to cooperate with the sender in order to establish the identity of the suspect. On the other hand, if the need arises to verify the chooser’s requests in a court of law, the proxy and the sender could work together to establish the identities of the records requested by the chooser. If an ordinary SPIR protocol is used to acquire investigative data this would shift the balance of the privacy protection from innocent individuals towards the suspect and the secrecy of investigation. Introduction of the semi-trusted third party into this protocol would restore the natural order, where the rights of the data-subjects, ordinary citizens, are put ahead of the secrecy of the investigation. This is likely to benefit the general public’s perception of PET based investigative systems.

CONCLUSIONS

This paper presented a platform for investigative data acquisition that preserves the privacy of the suspects and secrecy of the investigations. After a careful analysis of the related issues and research of available privacy-preserving primitives IDAP has been defined. The platform is based on PE protocol, a SPIR protocol based on commutative cryptography that allows retrieval of extra information about the records that are common between two datasets. Since, the features of the PE protocol closely match those required in retrieving investigative information form third
parties, only three improvements were required to form IDAP from this protocol. These improvements are the contribution this paper makes to the PIR domain.

In this paper a view that in certain circumstances, hiding the objective of the PIR protocol by running the data retrieval protocol against only a subset of the dataset provides sufficient privacy protection is presented. This is certainly the case in the investigative data acquisition process. The number of records that are collected per every interesting record is specified by the dilution factor $o$ introduced by this research. Since this factor can be changed before each protocol run, the investigators can dynamically chose the appropriate level of protection for the given investigation, the data subject, and the data controller. The protocol operates by creating a single encrypted table of identities held in the third party’s database and allowing the investigators to privately match their suspects against this table. Once the investigators know the encrypted ID of the suspect a number of records are selected at random to make up a request of size $o$.

Consequently the data controller can then facilitate private data retrieval operating on a small subset of the database. In this way the processing time is significantly reduced and requests from large databases are feasible. Such technique could be potentially risky if the same enquiry is made against few different data controllers, since the intersection of the requested results could help the cooperating controllers to identify the suspect. However, according to the Police it is not likely that data controllers will cooperate in such matters, especially if such cooperation would be forbidden by the letter of law. In the cases that the data is being retrieved from large databases that require use of the dilution technique during data retrieval process, the interesting records are usually identified by a mobile phone number, or an IP address. Phone numbers and IP addresses are often unique to the operators and their assignment can be obtained from the call and network routing tables. In this way, in most cases, the investigators only ask a single operator for information about a given identity. This fact makes most investigations equivalent to a single database PIR allowing dilution to be applied, with no adverse affect on the privacy of the data-subjects.

A technique that enables investigators to perform complex database searches maintaining privacy is also provided in this paper. The investigators can create a list of values for every secret input parameter, and use this list in the same fashion an identifier of the interesting record would be used in the ordinary PE protocol. Consequently, neither the complexity of the protocol nor its security properties are altered in providing this additional functionality to the acquisition protocol.

Finally, this paper addresses concerns of the general public in employing encryption based PETs to handle sensitive data. People generally trust the security process more that they trust encryption. For this reason a semi-trusted third party is added to the protocol to act as a proxy. The entire communication between the investigators and the dataholders is thus done via this proxy. The key objective of the proxy is to filter out the records that were not requested by the investigators. This protocol is secure only as long as the proxy is trusted not to cooperate with the dataholder. For this reason a party whose main concern is privacy of the individuals should hold this function. Therefore, in the UK, ICO could handle such a function. This approach ensures that the balance between the privacy of the alleged suspect and the privacy of the innocent individuals are maintained after IDAP is introduced as a data acquisition technique. Such move is likely to improve the public’s perception of the platform.
REFERENCES


¹ICO in the United Kingdom, is a non-departmental public body which reports directly to Parliament and is sponsored by the Ministry of Justice. It is the independent regulatory office dealing with the Data Protection Act 1998 and the Privacy and Electronic Communications (EC Directive) Regulations 2003 across the UK.