A multi-disciplinary framework for cyber attribution

Ronald Smyth
BSc (Hons)

A thesis submitted in partial fulfilment of
the requirements for the degree of
Doctor of Philosophy
at
De Montfort University

August 2017

Abstract

Effective Cyber security is critical to the prosperity of any nation in the modern world. We have become dependant upon this interconnected network of systems for a number of critical functions within society. As our reliance upon this technology has increased, as has the prospective gains for malicious actors who would abuse these systems for their own personal benefit, at the cost of legitimate users. The result has been an explosion of cyber attacks, or cyber enabled crimes. The threat from hackers, organised criminals and even nations states is ever increasing. One of the critical enablers to our cyber security is that of cyber attribution, the ability to tell who is acting against our systems.

A purely technical approach to cyber attribution has been found to be ineffective in the majority of cases, taking too narrow approach to the attribution problem. A purely technical approach will provide Indicators Of Compromise (IOC) which is suitable for the immediate recovery and clean up of a cyber event. It fails however to ask the deeper questions of the origin of the attack. This can be derived from a wider set of analysis and additional sources of data. Unfortunately due to the wide range of data types and highly specialist skills required to perform the deep level analysis there is currently no common framework for analysts to work together towards resolving the attribution problem. This is further exasperated by a communication barrier between the highly specialised fields and no obviously compatible data types.

The aim of the project is to develop a common framework upon which experts from a number of disciplines can add to the overall attribution picture. These experts will add their input in the form of a library. Firstly a process was developed to enable the creation of compatible libraries in different specialist fields. A series of libraries can be used by an analyst to create an overarching attribution picture. The framework will highlight any intelligence gaps and additionally an analyst can use the list of libraries to suggest a tool or method to fill that intelligence gap.

By the end of the project a working framework had been developed with a number of libraries from a wide range of technical attribution disciplines. These libraries were used to feed in real time intelligence to both technical and nontechnical analysts who were then able to use this information to perform in depth attribution analysis. The pictorial format of the framework was found to assist in the breaking down of the communication barrier between disciplines and was suitable as an intelligence product in its own right, providing a useful visual aid to briefings. The simplicity of the library based system meant that the process was easy to learn with only a short introduction to the framework required.

Acknowledgements

I would like to thank many people for their support and patience during my time as a student. Firstly I would like to thank Tim Watson for originally offering me the opportunity to take on a PhD and believing in my ability to take on such a challenge. I must thank and apologise to John North and Andrew Nicholson who taught me during the initial phases of the trials and pitfalls. I am certain it was emotional for all and their patience and friendship has been a constant bedrock. I would like to thank Andy Jones and Helge Janicke for assisting me during the final stages of process and providing the impetus to finally complete this never ending task. My family have always supported and believed in me, encouraging me from an early age to explore my interests which has resulted in my current passion, for which I love them. Finally and most importantly I would like to thank my long suffering wife for supporting throughout this endeavour for whom this would not have been possible.

Publications

The following papers were created and published or projects completed during the initial stages of this PhD.

* A multidisciplinary approach to cyber attribution was presented at the IAAC symposium 2017. IAAC (Information Assurance Advisory Council) is a specialist body of experts who aim to develop and advise policy recommendations to central government to ensure that the UK have a robust, resilient and secure digital environment. The presentation was well received and won the accolade of best presentation of the day. [170]

* Performed a live demonstration on BBC radio Leicester using the developed framework to highlight the security implications of an open social media profile. This broadcast is estimated to have been listened to by 116,000 listeners. Further details can be found in the experiments section. [168]

* Contributed to a horizon scanning exercise on behalf of the DCDC (Development, Concepts and Doctrine Centre), MOD. The DCDC underpins strategic force development of the MOD and directly feeds into policy developed by central government. The recommendations and report remain classified.

* Involved in the development of cyber intelligence capability for the MOD through a series of workshops. The unclassified outputs of this will be published next year.

* Presented to the HEA (Higher Education Authority) at the National Conference on the learning and teaching in cybersecurity 2016. I presented on my social engineering coursework as a case study for highlighting the threat of social media. [169]

* Worked with a company to develop a vulnerability scanning methodology for use in the assessment of the cyber security of fire detection systems. This work is currently under embargo by the company whilst the vulnerabilities discovered are fixed however it is hoped that this work will be published next year.

* A European funded project investigating the security vulnerabilities in automotive vehicles. The Trusted Software Initiative (TSI) assessment framework was applied by us to a concept vehicle. The

attack surface of a vehicle was analysed and a vulnerability framework was developed. Finally this framework was applied to a test vehicle and a number of vulnerabilities were identified.

* The threat actor framework developed within this document was first proposed in proceedings at the European Ground System European Workshop (ESAW) [147]. The framework was applied to the satellite industry.

* A highly classified project investigating and developing vulnerabilities and exploitations for specific hardware platforms. This work remains highly classified however went through an extensive peer review process with both the sponsoring company and DSTL.

Media Appearances

Date	Publisher	Role	Title
20th September 2017	BBC Leicester	Analyst for live experiment	The social media threat
18th May 2015	ITV (TV)	Technical Consultant (Uncredited)	Scammers
17th March 2014	Wired.com	Interviewee	GCHQ test budding cyber spies with stuxnet style crisis
12th February 2014	East Midlands Today (TV)	Interviewee	Police urge the reporting of cybercrime
November 2011	Digital Forensics Magazine, Issue 9	Author	Digital Archiving and Data Recovery
19th June 2008	Independent	Interviewee	Cybersleuths are in demand

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Statement

Within the western world, society has become reliant upon the internet and the interconnected world, sometimes referred to as 'cyber'. Billions of communications and billions of dollars worth of transactions are happening daily across this medium. It is estimated that 40% worlds population now has an internet connection, and there is an increasing reliance upon this capability. Some are even heralding this as the beginning of the 'information age'. Despite this reliance, the cyber world is very much in its infancy. The technology behind the cyber world is constantly evolving, developing new capabilities. As the understanding of the cyber world develops it has become apparent that securing the environment is of vital importance to its success. Within the UK Security Defence Review it classified the threat of cyber attack against the nation as a tier 1 threat, the highest possible grade alongside international military action. Currently the situation within the cyber world is that of near anarchy. With tens of thousands of strands of new malware are being released and hundreds of thousands of attacks being launched daily. The cyber space is an extremely hostile environment to work within. Currently within the Information Assurance world there a notion that you should assume your network is already compromised. The situation has been likened to that of the wild west. A major factor in this situation is that attribution within the cyber domain continues to be almost impossible, with criminals, terrorists and hackers able to carry out activity with relative impunity. As long as the attribution of an attack remains unattainable, the security of cyber space is unlikely to improve. Many different specialist are working on the attribution in a range of different fields in order to try and resolve the attribution problem in order to improve our capability to respond to a cyber incident. Unfortunately at present these specialisms work in isolation, often communicate in their own specific language and are incompatible with each other. There is no suitable platform for these specialist to share their information to provide a much richer attribution picture and hopefully resolve the attribution problem.

1.2 Aims and Objectives

The ultimate goal of the project is to create a framework suitable for the sharing of attribution data between technical specialists. The focus will be on cyber security and linguistic analysts. For the purposes of this thesis any characteristics which will assist in any element of the attribution problem will be considered. It is assumed that an attack has been identified and that the attackers actions have all been allocated to an individual attacker. The detection and allocation methods are outside of the scope of this body of work.
With a better understanding of what is possible to assist attribution within the full spectrum of cyber activities, an analyst is in a far better position to defend their own network, capture relevant data for potential evidence and put in place the tools with which to assist them in this role. In order to achieve this aim the following objectives have been defined:

Objective	Methodology	Chapter
Study the literature	Understand linguistic techniques	2
	Survey current linguistic attribution techniques	2
	Analysis of current tools	2
	Survey current technical attribution techniques	2
Analyse the gaps in current knowledge	Critical analysis of literature	2
Analyse suitable attributes	Survey current literature	2
	Reverse Engineer protocols and files	4
Develop a cyber attribution framework.	Waterfall development	3
Develop a Threat Actor framework	Waterfall development	3
Develop a library creation mechanism	Waterfall development	4
Validate the proposed framework	Perform experiments against aims	5

1.2.1 Literature Review

The literature review will be split into three area's. Firstly an understanding of linguistic analysis techniques will be explored. This will then be focused into the current practices around forensic linguistic attribution. Finally a in depth survey of current cyber attribution techniques. All attribution techniques will be critically assessed and suitable attributes will be extracted for use in the framework.

1.2.2 Develop a Cyber Attribution Framework

Having completed an in depth literature review of all material pertaining to attribution that could be applied within the cyber domain, a comprehensive cyber attribution framework will be developed. This will

encompass all potential avenues for attribution. The framework may also be used to highlight avenues that are currently out of scope of the industry. The framework must be robust enough to enable expansion as technologies develop, however accurate enough to be useful to analysts in todays attribution problem.

1.2.3 Develop a Threat Actor Framework

As discussed within the definition of attribution, there are different levels of attribution that are suitable for use within the cyber domain under differing circumstances. One of the levels of attribution is to classify a threat actor under a number of suitable categories. An analysis of current classification must be performed, as well as an assessment as to their usefulness. Once the current standards have been considered, and current practitioners have been consulted as to their preferred classification set, a complete set of threat actors, including definitions will be provided for use along side the Cyber Attribution Framework.

1.2.4 Verification of Frameworks

In order to prove the validity of the proposed frameworks a fully completed demonstration of the frameworks being applied to a set of test data must be completed and analysed against the defined success criteria.

1.2.5 Explore identified area's of weakness

Having completed the process of developing the Cyber Attribution Framework, it will be possible to highlight area's within the framework in which further research is required. Once identified, it will be possible within he scope of this thesis to explore one of these area's in great depth.

1.3 Novelty

The primary novelty is delivered through a framework which may be used by multiple specialist teams to develop an overarching attribution picture. The model can be used both in slow time and fast time investigation providing a powerful framework for the development of threat actor fingerprints, and ultimately this picture can continue to be built until the level of attribution has been acquired. This is the first time that experts from a range of disciplines have been able to share data to build an attribution picture within a mathematically bound framework. Whilst the framework would be suitable for any specialism, the proof of concept will be developed for cyber specialist and linguist who are already likely to be working together in an intelligence setting. The framework will also enable the rapid dissemination of complex data to assist other analysts or provide the basis for a briefing. Finally there is a clear methodology for the

creation of other data types to implement within the model, providing a completely extensible model for future works.

1.4 Scope

As explained in the introduction to attribution, there are several layers at which attribution can be performed. The focus of this work is at the Tactical level, providing a capability for analysts operating at this level. There will be some benefit at the operational level although this will be limited to comparative analysis. The model is not design for strategic analysis. Whilst the model is designed to enable all varieties of attribution capability to be used, for the purposes of proof there will only be a limited number of ”libraries” implemented. For the purpose of the proof of concept the libraries will focus on the technical skills of a cyber security analyst and a linguistic analysts. This is because they are likely to already be working together in an intelligence organisation and have an immediate requirement for this framework.

1.5 Contribution

There is currently no formalised system to enable the sharing of attribution data between multiple disciplines. Through the development of a model with a sound mathematical base it is possible for the first time to consider all attributes from all disciplines within a single threat actor picture. This will enable the future collaborative research from multiple fields into resolving the cyber attribution problem.

1.6 Success Criteria

The aim of this thesis is to improve on the current work flow of cyber threat analysts when dealing with a cyber incident, and enable them to consider all of the available evidence in order to obtain attribution to the attacker. It should enable effective communications between tools operators and analysts. If the model can be used to improve communications of attribution data, as well as assist an analyst in the creation of that intelligence product then it will be deemed a success.

The framework should:

- Be suitable to accept any form of attribution data
- Be suitable to perform live attribution analysis
- Be capable of highlighting intelligence gaps
- Enable fast communications of attribution data, no matter what the source

1.7 Thesis Structure

The thesis is structure to show the journey of researching current literature, developing a framework methodology, applying this methodology in the form of libraries and finally the testing of the framework. Chapter 1 will clearly define the problem domain, provide explicit definitions within the context of the problem domain and set the boundaries for the problem being explored. Chapter 2 will analyse all relevant literature relating to attribution within this context. It will be critically analysed in order to provide the best attribution methods to feed into the framework. Chapter 3 will be used to develop a cyber attribution framework, encompassing all relevant methods discussed within the literature review and highlighting area's that require further development due to the lack of literature. Additionally a threat actor framework will be developed to enable the labelling of identified individuals against their attributes to enable the prioritising of targets based upon capability. Chapter 4 will be used to develop a mathematical building blocks that overlay and feed into the framework. These will be in the form of libraries for individual attribution topics or tools. The process will be explored in full to enable the creation of new libraries by other experts. In chapter 5 a series of experiments will be performed to validate the framework against the primary objectives. This will allow for a critical analysis of the framework itself. Finally chapter 6 will draw conclusions from the process of developing the framework as well as the experimentation with a look towards future developments.

1.8 Conventions Adopted

In order to develop fully explore the arena of cyber attribution clear definitions must be provided. Consideration will be given to the issues around the current definitions in use within the field before defining the definitions that will be used throughout the document.

1.8.1 Cyber

There is a great deal of controversy over the definition of cyber. Despite the word being in common use there is no real consensus as to its meaning, and is often the focus of many heated discussions. The word cyber itself came into existence through the author William Gibson who first used the short novel "Burning Chrome". It was made more popular by his later book "Neuromancer" in 1984 which went on to describe cyber as;

> A consensual hallucination experienced daily by billions of legitimate operators, in every nation, by children being taught mathematical concepts. A graphic representation of data abstracted from banks of every computer in the human system. Unthinkable complexity. Lines of light ranged in the non-space(sic) of the mind, clusters and constellations of data. Like city

lights, receding.

Whilst this definition is interesting from an academic viewpoint, it is not useful for providing a definition for use within this thesis, or as it is commonly understood in todays society. Within the Collins dictionary it is simply describes cyber as a

combining form, indicating computers

[48]. This is a very simplistic definition, focusing on the computer element of cyber, which in my belief encompasses much more than simple computing. The Oxford English Dictionary agree's that there is a wider focus with its definition;

relating to or characteristic of the culture of computers, information technology, and virtual reality: the cyber age

[49]. This still is not a definition that can be used in this context, and remains computer centric. It is widely agreed that cyber encompasses more than computer systems, however there is debate around whether cyber is focused on the data, communications or the infrastructure. A lot of this debate is dependant on the focus in which you are considering the cyber world. For the purposes of cyber attribution, our focus is from that of cyber security. Although all play their part in the cyber world for the purpose of this thesis it was decided that cyber was information based. The use the word information is used specifically, because this could encompass not only the raw data, but also the meta data which, depending on the context can be equally as important as the data itself. The information must be in some digital format to be deemed cyber. This definition can be argued through the analogy of a book. A hard back book is of itself not cyber despite containing information. If however the contents of that book are transferred to an E-Reader then although the information that is contained in the book is the same, the very fact that the information is now digital makes it a cyber thing. The information can either be in a stored format or in transit through some transmission medium.

1.8.2 Cyber Space

Given the word cyber is relating to information, the communications of that information has to be referred to in some way. The phrase "cyber space" will be used in this context. Although this phrase was originally used as a slang term for the internet its official dictionary definition is "a realm of electronic communication" [49]. The realm can be through any medium (copper wire, fibre optic, Radio Frequency etc...) massively interconnected and be used to transmit or store any type of information. The underlying protocols do not matter as long as it is transferring or storing information and so it avoids the closed thinking that this only applies to computers and IP based networks.

1.8.3 Cyber Security

Cyber Security therefore relates to any action that is required to keep the information safe and available. The information may be in transit or at rest. For this definition security is defined under the widely accepted CIA triangle of Confidentiality, Integrity and Accessible [197]. The defence of the cyber space is as important as the information. Despite the information existing in its digital format it should not be overlooked that a physical attack on equipment can be just as effective as what is normally considered as the threat of attack through cyberspace.

1.8.4 Cyber Attribution

There are many definitions of attribution used within the cyber security environment [83] [196]. Ultimately it is the search for characteristics which can be associated with an attack or an individual. Different classifications of attribution are possible under this definition including;

- Classification of Attackers
- Action Assignment
- Traceback Attribution
- Full Attribution

1.8.5 Classification of Attackers

The Classification of Attackers occurs when you begin to classify attacks. Using a number of characteristics from an attack it is possible to classify an attack into a broad category. This can be done quickly with limited amounts of data available to you. For example you may look at the sophistication of an ongoing attack to decide on the actions required to mitigate the threat. Should the threat be considered a low skilled hacker using standard tools, the threat actor would likely be classified as a ”script kiddie”, and minimal resources would be allocated, however should the attacker be deemed highly skilled then additional resources will be allocated to mitigate the threat. At this point no attempt has been made to identify the attacker. This process can be useful for developing a standardised response to a threat, based on a category that they fall into and would form the basis of Standard Operating Procedures (SOP’s)

1.8.6 Profiling

Profiling can be used to attribute an action to a known individual or group. Whilst you might not know the identity of that individual, through attributing activity to that fits their profile, it is possible to detect all of

the activity associated with that attacker. This information can once again be used to develop a mitigation strategy. It is possible to gather profiling information about an attacker prior to an attack happening on your organisation, through information sharing partnerships [145] and active threat intelligence gathering [57].

1.8.7 Traceback Attribution

[196] defines traceback attribution as ”any attribution technique that begins with the defending computer and recursively steps backwards in the attack path toward the attacker”. The ultimate goal of traceback attribution is to follow the communications back to the original source, usually in the form of an IP address. This process is made extremely difficult due to the fabric of the internet and methods commonly employed by attackers to mask their IP address. Once an IP address has been obtained, it is possible with the correct authority to obtain a physical location associated with that IP address from an Internet Service Provider (ISP).

1.8.8 Full Attribution

Full attribution is when a specific action can be assigned to an individual person. This goes one step further than traceback attribution and attempts to put ”fingers on the keyboard” of the attacking machine. You are not only concerned with the machine which performed an attack, but who was using the machine at that time. This is the level of attribution required for legal proceedings.

For the purpose of this thesis any attribute that can lead to any level of attribution discussed above will be considered.

Chapter 2

Literature Review

2.1 What is Linguistics?

The field of linguistics is the attempt to understand language as a science. The results of this can be seen within a great number of independent research areas within the linguistics field. The field of linguistics is huge, and far to broad to cover in depth. This report will consider the wider field of Linguistics before focusing on the areas that are relevant to the question in hand. Although the areas of linguistics are presented here as individual elements, there is a great deal of crossover and cross pollination between the elements.

2.1.1 Phonetics & Phonology

The language the humans use are ultimately made up of a series of sounds at different tones and frequencies. The field of Phonology is study both how we create the sounds and how we hear the sounds [36]. There is the study of the physical elements of the sound creation, how we adjust our airflow, change the shape of our mouths, move the location of the tongue and use it in combination with our teeth as well as creating the sound through the vibration of our voice boxes. This is a very complicated mechanical task with a number of different elements interacting to create the sounds that we later interpret as words. There is also a sub-field focusing on the the different sounds that different languages use. There are a number of phonomes that are used within one language which are not used in another, making it difficult for a non-native speaker to use the language as the sound they have to create is very alien to them [107] [108]. An example of this would be the “ll” sound used in Welsh which is not used in English and is therefore very difficult for an English speaker to pronounce [142]. Additionally to the creation of the sound Phonologists will also study the structure and make up of the ear, and it interactions with the brain in an attempt to better understand how we receive and interpret the sound waves that are created. How do different sounds fit together to create words and are there any rules? How can we tell when one word finishes and another begins in an audio waveform? What is the process to then understand these waveforms as words? What is the rhythm to

speech and how do we tell when another person has stopped talking?

2.1.2 Syntax

The field of syntax is the study of the structure of a sentence and the relationship of words to one another [33] [123] [14]. It is concerned with how a sentence can be put together and the rules that govern that structure. In order to study the rules there once again have to be block elements with which a sentence can be built. To a syntactician the building blocks of a sentence are the words. These words are then classified in different categories such as Noun, Verb, Adjective, Adverb, Determiners, Complementizers, Conjunctions, and Prepositions [188]. It should be noted that a single word may fall into different categories depending on the context. Firstly you must come to an agreed set of rules as to what exactly classifies as a noun or a verb etc. In some cases these broad categories are broken down further in sub-categories, for example verbs can be split into ditransitive verbs, transitive verbs and intransitive verbs dependent upon how many nouns the verb appears with. Once you have the agreement as to the types of words, you can then begin to analyse how these words fit together within the confines of the language. For this a set of Generative Rules are created which will define the order in which types of word can appear. As an example of a we could state the a sentence can be created from a determiner then a noun and a verb (e.g The dog barked.). This can be represented as a rule statement $S \Rightarrow DETNV$. A huge number of Generative Rules would have to be created in order to fully document a language if they were used in this form and so specialist units are created which can be expanded upon. These specialist phrasal units include Verb Phrase (VP), Noun Phrase (NP), Adjective Phrase (ADJP), Adverb Phrase (ADVP) and Prepositional Phrase (PP). There is also the conjuncture rule which enable the use of the word "and" to link two phrases. With this additional set of rules, and the fact that they can be self referential results in every sentence within a language being modelled.

$S \Rightarrow NP\ VP$
$VP \Rightarrow V\ (NP)\ (ADVP)$
$NP \Rightarrow (DET)\ (ADJP)\ N$
$ADJP \Rightarrow (DEG)\ ADJ$
$ADVP \Rightarrow (DEG)\ ADV$
$X \Rightarrow X\ Conj\ X$

With all of these rules the elements can be expanded until you get to the base element of a word type. Elements that appear within brackets are optional and can be left out of a sentence. It should be noted that a syntactician can create sentences that do conform to these syntactic rules of a language but would not

be accepted as valid by the speaker of the language, which is an area of great interest in the field. The development of these rules forms the foundations of Natural Language Processing, in which a computer will attempt to apply these rules to assess the validity of a sentence, or create it own syntactically valid sentence.

The particular rule set described above is very simplistic but works for a number of languages, primarily English, however it will not work for all languages. Whilst every language is unique in some way, and as a result each language will have its own set of rules, there is a theory posed by Chomsky [35] [34] [41] that there should in fact be a universal set of rules which can apply to all languages. This is known as the Universal Grammar Hypothesis.

2.1.3 Semantics

The field of semantics is the field of meaning. This can be at word, sentence or text level. This is a far more philosophical area of study and includes fundamental questions such as ”what does it mean to be a “Man”? As a result of these fundamental questions the field include some of the great philosophers throughout time including Plato [182] and Saussure [45]. An alternative approach to semantics is the area of sentiment analysis, where there is an attempt to understand the feelings that the author is attempting to portray. This area of linguistics is not expected to assist in the area of cyber security and so was not expanded upon further.

2.1.4 Discourse Analysis

The field of Discourse Analysis is the study of stretches of language, larger than a sentence. This is broken into several sub fields including Pragmatics, Forensic Linguistics, Rhetoric, Stylistics etc. This can be in both written and verbal forms. The field of pragmatics is a study of how language is used in context to create cooperative conversation. The analysis is of a conversation as an attempt to understand how the information flows between participants, how you can assess when it is your turn to speak and an analysis of perspective. One theory of pragmatic study is that of Transactional Analysis [176]. Transactional Analysis states that the ego is in one of three fundamental states at any given time. These state are Parent, Adult and Child. In the parent state you would think act talk and respond as your parents did when you were a child. In the Adult state you are considered to view the world objectively, apply logic, and calculate chance based on your previous experiences. In the child state you respond with your feelings in a carefree manner. There are further levels to this model which expand upon the simplistic three layer model, for example there is the nurturing and controlling parent. People will naturally assume one of these roles dependant on the current conversation and their position within it. You can slip in and out of these states as the conversation develops.

Pragmatics is also interested in people with whom there is a pragmatic deficit due to an underlying condition such as Autism, Aspergers and Schizophrenia, in an attempt to develop techniques that would assist them in communication.

Stylistics is the study of style elements within a text [162]. It attempts to explain the choices an author makes and detect stylistic traits of an author. There is strong crossover with authorship attribution. Authorship attribution methods will be considered later in chapter 3.

2.1.5 Sociolinguistics

The field of Sociolinguistics is the study of how language is used within society [42]. It is especially interested in looking at how different social group use language differently, and how these differences have developed. Ultimately they are searching for what is causing the development and divergence in modern languages, and how language can best be used to interact with specific social group to achieve a desired result. From the perspective of my research if it was possible to identify speech communities using specific artificial language variants then that would assist a great deal in the attribution.

2.1.6 Psycholinguistics

The field of Psycholinguistics is an attempt to understand our processing of language and discover if the language that we speak does correlate to the way in which we see the world. This fundamental question is known as the Whorfian Hypothesis [198] [84]. This fundamental question is split into principles, that of linguistic relativity and linguistic determinism. The Principle of Linguistic Relativity is that the similar objects in a language will correlate and enable you to think of multiple objects with similar properties, however should a single object be stated then that similar objects would all be regarded as the same. It is all a matter of perspective when considering an object. The principle of Linguistic Determinism is that different cognitive styles between countries and races has resulted different development between an object and the word that we choose to use for it. As a result different countries consider different objects from a differing perspective and so will regard them differently. There is a further area of study around the brains ability to comprehend language and how the brain actually achieves its results. This is a subsection known as neurolinguistics. These areas were not expanded on in great depth due to the abstraction from the primary question.

2.1.7 Typology

The field of typology is performing classifications on elements of languages and attempting to detect similarities and differences in different languages. Once differences have been detected a qualitative study is

performed against a representative sample of languages in an attempt to discover how widely the differences are distributed. There are several sub-fields of typology covering the different elements that can be used to analyse a language such as Morphological Typology, Syntactic Typology [38], Phonological Typology Lexical Typology and Semantic Typology [120]. Some sub-fields are more developed than others [123] and the sub-fields do not have to be considered individually, for example some elements of language may be presented morphologically in one language but syntactically in another.

2.1.8 Historical Linguistics

The field of historical Linguistics is an attempt to understand the origins of language and how language has developed throughout the world, potentially from a single point of origin [28]. This field has developed a number of strands of research including attempting to define current languages in families, understanding the relationship between these languages and assessing how language changes over time. An example of the research in this area includes the analysis of the Proto-Germanic language tree which through several phases ultimately ends in the majority of the languages that are spoken in Europe today. Because it is believed that these languages have developed from a single root they can be considered "sister languages" having developed from a single "mother language". Because of the direct link between these languages similarities can be sought between them and traced back through older variants of a language. It is often possible to detect the introduction of new words to a political or natural event that has caused the migration of a large number of people [109]. Obviously this gets more difficult the further back in time you go as records are not as reliable.

There is the potential to use the creation of classification families to group programming languages, and map their development as they emerge from previous and potentially superseded languages. With this mapping in place it could be possible to pick up on programming traits from the superseded languages, or languages in the same family to that which is being used to write the program. This type of analysis could be performed against source code of a suspect program and there is the potential that we may detect traits of other programming languages within the source code. These traits will be specific to the author based on the authors previous experience. It may reveal other languages that the author is capable of programming, thus revealing something about themselves.

2.1.9 Forensic Linguistics

Forensic Linguistics is the study of language for the purpose of answering a question in law. There are several questions that could be asked of a forensic linguist specialist such as the attribution question, a country of origin, identifying memers of a group or detecting a forgery. The important thing is that the

answer is capable of meeting the rigorous requirements of court. One of the main focuses of Forensic Linguistics is the area of authorship attribution. It is often important as a point of law to determine the author of a specific piece of evidence, be it a ransom note, suicide note or a fraudulent document. The fields of handwriting analysis, document analysis and digital forensics can be used to enhance the legal argument, although are not strictly forensic linguistics.

Another area of forensic linguistics is the semantic analysis of points of law to assist in the definition of a law. Often a law is written in such away that it is open to a certain amount of interpretation. A Barrister may attempt to use their own interpretation of the law for the benefit of their client. In this situation a Forensic Linguist may be called upon to fully consider the meaning and provide an interpretation for the court to use [136]. The is also the question of the "Language of the law". This is distinct from the interpretation of law and is in fact the analysis of power dynamics within a legal environment and how language is used by all parties. There is a particular concern in this environment that the power and language used is unfair on vulnerable witnesses, child witnesses and asylum seekers [136]. There is also the question of your rights to silence, the impact that has, and the effects of surrendering your rights to silence at specific times [136]. The area of authorship attribution falls in line with the attribution problem experienced within cyber security.

2.2 Authorship Attribution

Authorship attribution is the sub field of Forensic linguistics in which you attempt to analyse a text for the purpose of extracting unique features which will aid in the identification of the author. There have been huge successes within the field and it has come to be recognised in its own right as suitable for use within court.

2.2.1 Early developments in Authorship Attribution

The earliest developments in Authorship Attribution were entirely qualitative in their nature. Linguist would manually analyse texts, selecting features that they believed would best distinguish between a small number of candidate authors and then run a comparative analysis on the questioned work against texts known to be by the candidate authors. Because of its qualitative nature, it was an extremely time consuming process which heavily restricted the number of features that could be analysed, the number of texts that could be analysed and the number of authors who could be considered. The result of this was that only a handful of studies were performed, and almost entirely against well known works of literature which were anonymously authored or believed to be authored under a pseudonym. One of the earliest documented examples is Mendenhall who in 1901 performed a stylometric analyses of the purported works of Shakespeare, comparing the features of the authors Shakespeare, Marlowe and Bacon [199]. Mendenhall

discovered that there was a close correlation to the works of Shakespeare and Marlowe, however the works of Bacon were distinctly different. The key feature in his analysis was the word length frequency, although this method has now been discredited [90]. In 1932 Zipf made a phenomenal breakthrough for the field of linguistics when he discovered that the frequency of any word is inversely proportional to its ranking within the frequency table [210]. Put in other words, the second most frequent word in a text will appear half as often as the most frequent word. This was tested by Zipf against English and Chinese [210], but the discovery has now been tested against a large number of languages and the findings continue to be true. As a result Zipf's Law has become known as the universal law of language. In fact this law continues to be true in all manner of fields beyond that of linguistics including the scale of cities [85], the popularity of internet pages [150] and the scale of the Internet networks [5] The law is so ground breaking that it continues to influence linguistics to this day [82] [10] [157]. The findings were so influential, that Zipf's contribution to linguistics has been compared to Newtons Contribution to Physics [11].

2.2.2 The Computerised Revolution of Forensic Linguistics

As computer developed they were able to assist in the laborious task of what had been the manual processing of feature data. This assisted the linguist immensely, and whilst the process still remained a qualitative study, larger data sets were able to be compiled and used. This was the birth of "non-tradition Forensic Linguistics". This enabled the use of more statistically complex calculations to be performed and a wider range of statistical techniques to be used. It was during this time the Monsteller and Wallace performed their ground breaking study of the Federalist papers. Even with the aid of modern computer the analysis still took 3 years [166]. They did however successfully attribute authorship over a disputed set of the letters, which was independently verified via another method. This was the first time that a real linguistic problem had been verified like this, which The Federalist Papers remain a valid challenge in linguistics and are often used as the test case for new techniques [200] [78] [97].

As computer power developed still further, larger corpus could be collated and entirely autonomous systems could be developed in an attempt to remove the human from the process. It is fair to say that some of these statistical methods were not fully understood whilst being used leading to a number of false claims during their early development[151]. Despite this bumpy start there are a large number of Artificial Intelligence techniques which have been accepted. These highly complicated but successful techniques make up the bulk of current Linguistic analysis and is an area of continual development.

2.2.3 Authorship Attribution Questions

There are a number of questions of law that might be asked of a Forensic Linguist with regards to Authorship attribution. Each type of question asks for a different type of result but uses the same skill set of the Forensic Linguist. These questions can be broken down into a set of three fundamental question classes. These classes are that of profiling, the needle in the haystack problem, and verification [100]. The question of profiling is given a text or a set of texts, what information can you extract about the author. These profiles can be looking at the properties of an authors style which may then be used to identify the authors other work. Alternatively they may be attempting to infer deeper meaning about the author such as gender[75], political stance [60] or social status [42]. The needle in a haystack problem is, given a large corpus are you able to extract all of the texts from a single author, or cluster individual authors together. This is achieved through the careful selection of feature sets and has been proven to work on a number of occasions across a range of media [99] [102] [101] . This has been the focus of academic non traditional linguistics. The features that work are highly dependant upon the corpus itself, although an incorrectly used corpus can reveal false positives [151]. Once the clustering has taken place you may then wish to introduce a questioned text or a test case to see within which cluster the document falls. The question of verification centres around a single unknown. You are posed with a single document. Are you able to assign the unknown document to an author out of a set of suspects with a degree of confidence? This is the most practised element of forensic linguistics as it has practical application to the court. This is the question regularly posed by the court when considering suicide notes, ransom notes or forgery cases.

2.2.4 Elements of the Authorship Attribution problem

The authorship attribution problem can be broken down into two distinct area; Feature selection problem and the Analysis problem. Whilst the areas in of themselves are distinct, the problems can not be solved in isolation. The feature selection has direct impact on the types of analysis that are possible. Also the features themselves can be analysis specific. First we will examine each problem.

2.2.4.1 The Feature Selection Problem

Language is built of highly complex structures as explored earlier in this paper. Within these structures there are a number of differing features. These feature have been classified into a a number of feature sets, dependant upon the linguistic element that is being focused upon [175]. In total around 1000 features have been considered, with varying degrees of success [151]. Any of these features may be used in an attempt to distinguish different authors. The features selected depend heavily on the corpus itself, as only a subset of the available features will be suitable for the types of work within the corpus, or selective enough to detect

the variation between authors. Some feature may be suitable in one case but not in others, and as such there is no specific feature set that will work in all cases. To help enable the selection of features a framework has been developed [90] and modified to suit the needs of the corpus [207]. The current framework use for authorship attribution is split into Lexical, Character, Syntactic, Semantic and application specific features as described below;

2.2.4.1.1 Lexical Features Lexical Features are those that are made up of the lexical elements of the text. Examples would include word length, sentence length, vocabulary richness, word frequencies, word N-grams and errors.

2.2.4.1.2 Character Features Character Features are analysing a text at the character level. Examples would include character types (letters, digits, special), character N-grams and compression methods. The advantage of these features is that the techniques tend to be language independent [96]

2.2.4.1.3 Syntactic Features Syntactic features are focused around the structure of the sentence. These are the most common methods in Natural Language Processing. Examples of syntactical features include Part-Of-Speech (POS), chunking, Sentence and Phrase structure, Rewrite rule frequencies and error detection.

2.2.4.1.4 Semantic Features Semantic Features are focused on the word meanings. The analysis of synonyms and semantic dependencies can be used to analyse these features.

2.2.4.1.5 Application Specific Application specific features are those which are specific to the medium in use. Features include structural features, content-specific features and language-specific features. Without a specified medium these features remain undefined. For example within the medium of a word processed document, structural features can include the font used, the font size, any font colour changes, line spacing decisions, the number of spaces after a full stop etc. This is often the first area of exploration for a new medium [207] [155]

2.2.4.2 Summary of Features

The choice of what features to include and the bias that should be applied to specific features has a significant impact on the outcome of any analysis. Also with such a great number of features the computing power required to process all of these features increases exponentially with every feature added. Rather counterintuitivly, some features that you would expect to be important are found not to be, where as features that some would never consider have been found to be very successful in distinguishing authors.

As a result careful selection of a subset of all features available is required to provide enough determining factors to usefully be able to distinguish the texts. The ultimate aim of the feature selection problem is to attempt to work out which features within a document will provide the greatest indicators of an individual author within the corpus. The best selection can only be sought through experimentation. With regards to artificial languages, the structure is less free form due to the application of protocols, however the content of the individual message can be more diverse that natural language as multiple programs will use the same protocol to transfer messages and those messages are likely to be program specific. Additionally to this there are a number of optional features that a programmer can choose to use as a design choice. The framework that has been used here is easily modified and should be considered if developing a cyber security solution.

2.2.4.3 The Analysis Problem

Once the features that an analyst is going to use for the distinguishing of documents have been determined the method must then be determined. As stated previously the choice of what features to include and the bias that should be applied to specific features has a significant impact on the outcome of any analysis. The allocation of bias forms the basis of the analysis problem, with analysts naturally choosing features that support thier argument. With a large number of analysis techniques at the disposal of forensic linguist, the selection of the technique becomes non trivial. The selection will be based upon several key factors including the type of question being asked (Profiling, Needle in the haystack, Verification), the corpus itself and the numbers and types of features that have been selected. There is then the additional problem of the time it takes to perform the analysis as the greater the number of features the greater the processing power required to perform the analysis. Luckily and rather counterintuitivly, it can often be better to have fewer features that produce the best results [112] [64]. Having obtained values for the features that the analyst has chosen, a desicion must now be made by the analyst as to the best method for analysis. These include N-Gram methodology, Back Propagation, Decision Trees, Support Vector Machine and vector space.

2.2.4.3.1 N-Gram method using Hidden Markov Modelling The N-Gram methodology is given a known state, what do you expect the next state to be. For the purpose of linguistic analysis the state will the the state of a specific item within the sentence. Items in the model can be characters, words, bytes [62] or any other definable section. N-Gram methods are split into the number of items that they take into account. For a uni-gram model only the current item (X) is considered, and the probability of all following options are calculated. For a Bi-Gram model the state of the current item, and the previous item are considered X, X_1. A Tri-Gram model therefore takes account of the two previous states. (X, X_1, X_2). The items must be classed, so in the case of words they can be classed using syntax modelling. The model will then use a training corpus to assess the probabilities of given a the current state to the N-Gram depth, what is the

probability for each type of item. The advantage of this method is that it can be used to process a text of any size (or even a data stream) and continually produce output. The Hidden Markov Modelling element enters when calculating the probabilities. There is an assumption with HMM during training that all of its probabilities have been estimated high. These probabilities are accurate for the training corpus, however they are likely to be an over estimate when you consider additional texts. As a result a probability mass α is removed to enable the model to handle use cases that appear in the actual corpus that did not appear in the training corpus. This affects all elements of the model equally and thus does not change the overall probability shape. There are a couple of ways to calculate α using either linear interpolation or Katz back off modelling [94]. A further advantage of HMM is that it provides a value of perplexity. This is a measure of how good the language model is, with the lower value representing the best fit. Goodman performed an analysis of HMM against a vocabulary of 50,000 words in an attempt to distinguish the differences in perplexity given the complexity of the HMM. For a uni-gram HMM the perplexity was found to be 955, with a Bi-Gram HMM the perplexity was 137 and for a Tri-gram HMM the perplexity was considerably better at 74 [66]. At this point it was found that going beyond the tri-gram produced diminishing returns whilst massively increasing the processing required. The vast majority of Natural Language Processing sticks with tri-grams. This forms the fundamental method of Natural Language Processing.

2.2.4.3.2 Back Propagation Back propagation is a training method for Neural networks in Artificial intelligence. It is a process of allowing a neural net to make guesses at an answer and then feed back whether those guesses were correct or not. You would use this form of training when you have a corpus of known authors and texts, and the neural net would apply its own weighting to the functions set to come up with the best weighting for the corpus. You are then able to introduce an unknown text to the neural net and it will use its now defined weighting to classify the document. There is a weighting-synapse value which is set to enable you to decide how quickly or how well the neural net learns. This weight-synapse value is automatically adjusted by the neural net, such that the best results change the least, thus the neural net is always tending towards a better answer.

2.2.4.3.3 Decision Tree's The decision tree method is based on a training corpus. What is the probability of a specific branch being followed given the current position. This best falls in line with a syntactical analysis whereby the specialist phrasal rules form the tree's, and an author personal preference can be deduced through a training corpus. Once you have the probability of author author using a specific route through the tree this can be compared against the questioned document and a comparison can be made.

2.2.4.3.4 Support Vector Machine Support Vector Machines are an artificial intelligence method used for defining the best boundary between two classes. They have the capability of performing both linear and non-linear classification. The ultimate aim for the decision boundary is that it maximises the distance to each class, whilst ensuring that all of a class remain the correct side of the decision boundary. This process is repeated for each feature resulting in separate SVM's for each feature and author pairing. This results in a large number of SVM's the results of which can be combined. You can also apply weightings to each SVM/feature as a part of the combining, highlighting the features which are the most important in case there is disagreement when the test corpus is introduced. Once the best boundaries have been found for all classes, the the questioned text can be introduced to the SVM's to see which class they sit within. These results are captured from all of the SVM's and the processed to give the result for the most likely author. SVM's have been used in the authorship attribution arena, with Abbasi [4] stating that SVM's generally outperform other methods of machine learning.

2.2.4.3.5 Vector Space Vector Space is a further artificial intelligence technique designed to create clustering. Rather than consider each feature on an individual plot it creates a multidimensional space of all features and plots texts within this space. Once again individual authors texts should cluster together. One of the main issues with this technique is being able to visualise the results. If you chose to use 30 distinct features then you have a 30 dimensional space which has to be represented somehow. The method has however been successfully used for authorship attribution [91].

2.2.5 Summary of non-traditional Authorship Attribution

The Function Selection Problem and Analysis Problem are two highly complicated issues in their own right. The complication is compounded by the fact that the problems are not independent of each other. Good understanding of the available features, the classification of features, and the analysis techniques that are suitable for those feature sets are required to provide an informed decision. Even then experimentation is required, as the results can be counter-intuitive to logical thought.

2.2.6 The traditional approaches to Authorship Attribution

As has been shown the Authorship Attribution problem is multifaceted and extremely complex. so far only non-traditional automated methods of clustering authors based on a feature set have been considered. There are however a number of traditional methods (some of which now can be computerised) which take a subtly different approach to resolve the problem, or at least part of the problem. A linguist may use these methods for authorship attribution or to attempt to extract information about the author. These methods are considered below.

2.2.6.1 Vocabulary

The selection of vocabulary used is often a fantastic indicator of an individual author. Any author has completely free choice as to the words that they choose to use. There are a variety of techniques that can be used to give an indication of the author. These include available lexicon, localised variables in vocabulary, learning styles and vocabulary richness. It should be noted that there are a number of limitations in only using the vocabulary and as such this should be used as only one indicator. Unfortunately the use of vocabulary is very easy to falsify if you have a sample of someone else's work. Specific examples of how vocabulary can be used are given here.

2.2.6.1.1 Available Lexical When examining historical documents, it is clear to see that the use of language and vocabulary in use is in a constant state of flux. Words go in and out of fashion and new words are created. Through analysis of the vocabulary used in a questionable document it is possible to compare this with the language in use at the time the questionable document is reported to have been created. This method can be used to help validate the authenticity of a document. A great example of this is the "Beale Letters" [106]. In this example a document purporting to be from 1822, published in 1860's, giving the directions to a hidden treasure included words that were not included in the Oxford English Dictionary until 1844. There were a number of other words used which had gained popularity by the 1860's but were not in common use during the 1820's. With regards to our specific problem domain it may be possible to detect the version of software being used due to the functions that are available within that piece of software. Using this method we could give an earliest possible date for a piece of software.

2.2.6.1.2 Localised variables in Vocabulary Further differentiation of vocabulary can be discovered in the analysis of certain words [17] [139]. Specific areas of a country use different words to signify the same thing. These local variations can be used to narrow down the likely location that an author has come from. As an example of this the words used for a bread roll vary significantly over a relatively small area. The terms include Cob (Midlands), Bun (Northern England), Bap (Scotland/Ireland), Barm (Manchester, Liverpool and South Lancashire), Batch (Wirral, Atherstone, Nuneaton, Bedworth and Coventry), Bin Lid (Mersyside), Blaa (Waterford, Ireland), Muffin (Rochdalem Oldham, Bury and Ashton-Under-Lyme). Should an author choose to use a word with a localised variable they are releasing information about either their current location or where they were raised [17]. This type of analysis was used in what is arguably the first Digital Forensics case. Cliff Stoll detected a hacker had breached Berkeley National Laboratories computer systems in late 1986 [177]. He worked closely with the authorities to develop a number of techniques in an attempt to track down the hacker to a physical location. One of his methods was simply to print out all of the commands that the hacker used whilst connected to his computer. It was noted that

the hacker was using the incorrect commands for the version of Unix that was installed on the computer. This version of Unix was developed by Berkeley itself and was used extensively on the West Coast of America, however this was the only area in which it was used. Other variants of Unix were used elsewhere in the world which used subtly different commands, and these where the commands that the hacker was attempting to use on the Berkeley machine. From this Stoll and his team deduced that there hacker could not have been local to Berkeley, or indeed be anywhere in West America otherwise he would have known and used the Berkeley variant of the command. Whilst just this information on its own didn't give away the hackers location, it did provide a key indicator as to where the hacker was not.

2.2.6.1.3 Learning Styles There is a good body of work that states that your choice of language is closely tied to your own personality type [137] [122]. The best example of this is that you tend to refer to situations based on your preferred learning style. In the examples given by Mann your learning style can be split into Visual, Audio or Kinaesthetic. Your preferred learning style will come across in your choice of words used to describe something or a situation. For example if you are a kinaesthetic learner you would use phrases such as "feels good to me", where as an auditory learner would use "sounds good to me". The Visual learner is more likely to say that the same thing "looks good" to them. A person will naturally default to their preferred learning style and therefore give away subtle information about themselves. There is a small potential that this sort of information may be leaked in comments or variable names chosen in a program.

2.2.6.1.4 Vocabulary Richness Vocabulary richness is a formal measure of how many different words are used within a text. The belief is that an author has a set vocabulary of known words, and a subset that the author has a preference for using. This subset, and the frequency in which each word within that subset is used is believed to be unique to the author [77].

Although on the surface this seems like a very simple measure there are a number of issues that quickly arise. Firstly the length of the text in question will greatly effect the vocabulary richness. A shorter text does not have the scope for the expansion of areas in which new vocabulary may be introduced. In an attempt to keep things succinct an author will default to more simplistic language to avoid having to expand in explanation.

The subject matter of the text will greatly impact the vocabulary used. Each subject has very specific vocabulary associated with it which would only be used in the context of that subject. For example different sports have their own vocabulary. In snooker it would not be out of place to refer to cue's, baize, potting, screwing, cushions, spin, or kick. With the exception of the final word on the list, none of this vocabulary would appear in a conversation about football. The word kick means two completely different things in

these contexts. In football it is the act of moving a ball with your foot. In snooker it is a term used to signify a bad contact between the cue ball and the object ball. This means that both the subject matter and the context in which a word is used is important for vocabulary richness analysis. The type of document that is being has an massive effect on the language use. For example a broadsheet newspaper article would be expected to use formal language to relay information where as an email between friends is far more likely to include slang and informal terms. A letter to that same friend may be a slightly more formal affair and so once again use a more formal choice of language. Finally an instruction manual, which is attempting to relay the same information will be written using a different vocabulary depending on if the manual is going to an engineer or an end user. The engineer would be expected to know and understand the specialist language associated with a device, where as an end user is not a specialist and so would not be expected to know the specialist language, thus that language is fully explained or completely removed. Here we can see that the type of document, the method of delivery and the expected audience have an effect on the vocabulary usage. A further issue in regards to vocabulary richness is do you consider all of the differing uses of morphemes. Does dance, dancer, danced and dancing count as four separate words, or are you simply using the same root word in different Contexts. Peng et al [140] have explored N-gram analysis at the letter level in an attempt to recognise this type of word usage.

All of these complexities have to be considered in both the analysis of a text and the selection of a corpus. As a result of these complexities no less than fifteen different methods for measuring vocabulary richness have been suggested [68]. Ultimately all of the methods attempt to formulate the number of words in a text (N) with the size of the vocabulary in the text (V). Various arrangements around this central theme have been suggested in an attempt to negate the effects mentioned above. The initial method was proposed by Yule [204]. His suggestion was that the analysis should focus on the word repetition rate, with the theory the more that a word is repeated, the smaller the vocabulary of the author. This method was found to be suitable to distinguish between two authors and associate an unknown text to one of those authors. This process did not work when there were large sets of authors [181]. A full an in depth analysis of these differing formulae was performed by Tweedie and Baayen [184]. They state that the formulae fall into two categories, Vocabulary Richness and Vocabulary repetition. There are pro's and con's for each approach depending on the wide range of complex issues as considered above, however they have developed a method of constructing confidence intervals for each method.

2.2.6.1.5 Hapax Legomenon A hapax legomenon is a word that appears only once in an entire corpus. There have been attempts to use hapax legomenon a features for authorship attribution. Hapax Legomenon are actually relatively common [90]. One attempt to use this feature focused on the number of hapax legomenon that appeared per text. The concept here is that an author with a wider vocabulary (subject and

period accepting) will produce a higher number of hapax legomenon per text. This number could then be used against an unknown text to see if there is any correlation [183]. Whilst the process did work in this case, it also highlighted the wide variation within an individual author.

The placement of a hapax legomenon has been considered. They regularly appear late in a sentence. This was hoped to be used to distinguish authorship [127] however this has since been disproved [167]. A further area of study is that of Dis Legomenon. Dis Legomenon has all of the traits of Hapax Legomenon however it is the study of words that appear exactly twice in a corpus [81]. The result of this is that it has all of the same problems as Hapax Legomenon except in the specific use case when the Dis Legomenon appears in only the questioned text and the most likely candidate author.

2.2.6.2 Spelling

The way in which a person chooses to spell a word can give information about the author. Primarily in this sort of analysis we are looking for errors that an author has made, or some form of unique spelling which could be specific to a country or region. Ultimately we are looking for an author to use a unique spelling in a consistent way.

There are a number of words in the English language which have multiple spellings. The difference in spelling could be a personal choice or more likely is an indication of country of origin. An example of this would be the word *"colour"*. Within the United Kingdom it is spelt *"colour"* however in America it is spelt *"color"*. You could however expect to find that a website developer within the United Kingdom uses the American spelling as this is what is used in the programming of HTML code which would be a useful personality indicator if you had traced the connection to the United Kingdom.

Whilst the above is a good example of gaining information about an author it can not be used to identify an individual. It may however be possible to identify a word that an author consistently spells incorrectly. Wellman [195] used this method to great effect live in court under the guise of a handwriting test to prove that the witness always spelt the supplied word in the same incorrect manner.

This field has become somewhat more complicated with the rise of the word processor as auto correct functions and spell checkers can now seriously affect the outputted text [99]. There are however situations where a spell checkers will fail to detect mistakes as a correctly spelt word has been used instead of the correct word for the sentence. Additionally the incorrect dictionary can be loaded causing the spellings to default to another country (US English instead of UK English). It may be possible through careful analysis of the outputted text to be able to determine the dictionary and/or program that has been used to create the text.

2.2.6.3 Structures and layout features

The structure and layout of traditional works were the decision of typesetter at the print works. As a result traditional works can not be analysed in this manor. The invention of the typewriter allowed the author to make some, albeit simple, formatting options, leading to individual styles being developed by an author. These features include post full-stop spaces, paragraph spacings and margin widths. It was not until the modern word processor however, that the author was given complete freedom of layout design. Now the author has free range to change fonts, text size, use of bold, underlining, italics etc. With such a wide range of options available to the author you can expect a much wider variety in features used, and thus a much greater chance of detecting a unique layout feature. These feature sets extend not only to typed works within a word processor, but additionally to structured E-mails , blogs and web postings [207].

A further area where structure and layout can be extensively analysed to detect unique authorship features is the area of software. The functions, commenting and design of a computer program is very much an art, with each programmer developing their own techniques and styles. As a result if you are able to obtain the original source code, these authorship choices become invaluable in authorship attribution. Unfortunately in a cyber security environment it is highly unlikely that you gain access to the original source code, and you are usually left reverse engineering a binary file. This is a file which has gone from the human readable form (source code) through a compiler and into a machine readable format (binary). In this process a large amount of the human element is stripped out of the program. As a result the selection of features for Binary code analysis is extremely difficult [62].

2.2.6.4 Metadata

One area that should not be overlooked when considering the analysis of documents is the metadata that is contained within a digital file. This falls firmly under the bracket of Digital Forensics and is a well developed area. When any file is created on a computer the file is given a number of attributes such as time/date created, modified and accessed, the user who created the file and the computer used to create the file. Depending on the type of file that has been created there may be a large amount of additional data associated with the file. A good example of this is a picture taken with a digital camera will include the make, model and serial number of the camera used to take the photo as well as information about the lighting conditions and any accessories attached to the camera at the time the photograph was taken. In regards to Microsoft Word documents there is a large amount of undocumented metadata that is captured when the file is created or modified, such as the last 10 users to modify the file and large amounts of the deleted data that the user believes has been removed from the document. In Microsoft Office 1997 it has also been found that the MAC address of the originating computer was hidden within the file which would

identify a specific computer [156]. Whilst these methods are not examining the natural language used by an author they do give a number of unique indicators as to who that author may be.

2.2.6.5 Synonym pairs

A synonym pair is a position in a sentence where more that one descriptive word could be used, and the author has had free choice as to what they use within their sentence. This is usually as a result of biases in their own vocabulary, although there can be external factor that influence their choice, such as having just seen the word in a plagiarism case. The switching of the word must not change the meaning of the sentence in any way, so the words must be completely interchangeable. An example would be the words "big" and "large". As a result of the authors free choice this enables synonym pairs to be used as a feature. An examples of this method include Ellegaard [53] who successfully used the method to distinguish authors of the Junius letters.

2.2.6.6 Function Words

In the previous example synonym pairs were used to compare authors. A similar technique also works for the function words within a sentence. Often function words can be interchanged without the changing of the meaning of a sentence. This results in the same technique of analysis being applied to function words. A good example of this being used is Monsteller and Wallace's [129] analysis in the federalist papers. The authors attempted to perform a synonym pair analysis however found that there were not enough synonym pairs for it to work. They carefully considered other options and decided that function words would work in a very similar manner to synonym pairs. With the addition of function words there were enough distinguishing features to distinguish the four potential authors, and the correctly assign the questioned texts. The results were verified independently through another field, giving good scientific backing to the discovery of the technique. As a result this was one of the most influential papers in Authorship attribution, and the federalist papers are still regularly used in linguistics today as a good case study to prove a technique. The function words themselves are words that carry no inherent meaning of themselves but play an important role in the grammar of a language. These function words are split into types of function words; Adverbs (Again, Ago, now, not, often, where etc.) Auxiliary verbs (are, being, can, they, you etc.) Propositions and conjunctions (About, after, although, his, its, something etc.), Determiners and pronouns (a, any, its, someone, something, that etc.) and finally Numbers (one, billion, eighteenth etc.). Relating this to the field of cyber security, functions play a critical role in programming languages. Although functions in this sense are entirely different to function words, there are a handful of functions that are interchangeable but do not change the overall workings of a program. If you have access to the original source code this could prove a useful, although somewhat limited, indicator of authorship.

2.2.6.7 Gender Detection

There is a belief that male and female authors write in in a tone that is specific to their gender, and as a result if an author writes a piece purporting to be the other gender the reader has an almost innate ability to detect this mismatch. This is most commonly reported in works of fiction where a male author is writing the dialogue of a female character, and female readers of the piece are able to detect subtle discrepancy's that lead them to believe that the author is not quite on the same wavelength [118]. There have been a number of attempts to perform comparative studies to establish what the differences are [75]. So far these studies have been inconclusive and there are some studies to suggest that it is a fundamental difference in the way in which the male and female brain works [163] , where as others state that it is due to males and females developing at different rates during their early language development [39] [71]. This is clearly a complex area that is not yet fully understood. Although this is an area of development it is not yet mature enough to use practically for our purposes. It may be worth revisiting in the future.

2.2.6.8 Learning Language

As stated previously, the development of language happens throughout your life, however the most rapid development happens during childhood [21] [123]. This natural language development is influence through a huge variety of interactions, from face to face conversations, presentations, TV and Radio broadcasts, telephone interactions, reading etc. With such a diverse number of inputs it can be difficult to detect the direct influences on a persons natural language. Contrary to this, in the learning of a programming language there are relatively few inputs available. As a result it should be possible to detect direct lines of influence when analysing source code. This influence is likely to have come from either a teacher with their own style, a system that the author has figured out by themselves that has worked in the past and so been repeated, or more commonly you get "Google coders" who take snippets of code from examples posted on the Internet. With this much reduce input set it should be possible to detect the influences from each of these vectors from an authors original source code. Whilst there is research around the detection of plagiarism within coding [37] and authorship of source code through linguistic analysis [61] [63] [27] and a separate body of work on the best pedagogical practises for teaching programming the author is unaware of any research into the linking of these two fields. Whilst source code attribution is an associated area to the attribution of an attacker this is not the primary focus of the research and so will not be considered further.

2.2.6.9 Forensic Voice

The field of Forensic Voice was briefly considered as potential avenue for research. This is the field of analysing the voice of a suspect on a recording and attempting to match it to the perpetrator of a crime.

There are essentially four primary methods that are used in this field. These are Phonetic analysis [126] [149], Spectrographic Analysis, Acoustic-Phonetic Analysis and Automatic Processing. Phonetic analysis is a process of assessing how the words are being pronounced to attempt to detect key words that have a distinct pronunciation that can be used as an indicator to single out a suspect. These inconsistencies are usually caused due to an accent or local dialect. A trained phonetician is able to use this method to highlight specific words used on the recording and compare them to a suspect with a reasonable degree of accuracy [149]. Spectrographic analysis is a method of converting the sample into a time/frequency grayscale diagram known as a voiceprint [95]. This method of analysis has now been discredited and is no longer accepted as a valid method for use in court [126]. Acoustic-Phonetic techniques combine the two previous methods, using the trusted phonetic analysis method but adding additional depth to the analysis by considering the spectographic analysis. Finally the Automatic Processing is using Signals Processing techniques to provide a useful characteristics in the voice patterns that can be analysed. These methods were discounted primarily because the methods are all based around non-discrete values caused by the waveforms of the voice. It was decided not to continue with this area of exploration as it would be pertinent to simply apply suitable Signals Processing Techniques, rather than go through the unnecessary step of considering a voice pattern for a computer communication. As we are dealing with discrete values with regards to the digital communications, these methods were not considered any further.

2.2.7 Summary of Authorship Attribution

We have explored the origins of authorship attribution, expanding into the modern techniques that are currently used. We have considered the three fundamental question types of Authorship Attribution and the two distinct problem sets of feature selection and analysis, considering each method in depth. What is clear is there are a huge variety of techniques available and the selection of which is best for a corpus, feature set and authors is a non trivial task. Both the Feature problem and Analysis problem need to be considered concurrently further complicating matters. Fortunately to assist us with this task Juola has developed a framework for comparing linguistic techniques [93]. A serious concern is that the majority of the techniques being employed within non-traditional Authorship Attribution are simply applications of Computational Intelligence Models. There is a realistic possibility that some of this work is being duplicated as it is being applied to other problem sets.

2.3 Application of Forensic Linguistic Techniques to other field

It has been seen there are clear indications that a number of Authorship Attribution techniques are highly successful in the literacy world, with demonstrable results. These methods should be suitable in any domain

where there are measurable features that can then be analysed. the application of these techniques to the problem domain of cyber security is not trivial. There are a number of examples where Forensic Linguistic techniques have already been transferred and used in other fields with some degree of success. Some of these examples have a direct bearing on cyber security. These examples could prove useful in the application of techniques, and may highlight issues the may be encountered.

2.3.1 Emails

A large number of the techniques that have been discussed in this paper are now being applied to emails. Although this can be for authorship attribution purposes for court [46] [136], the vast majority of this effort is being used for the purpose of SPAM filtering. SPAM is a process whereby billions of emails are created by computers either in an attempt to advertise a product or service or, more malevolently, cause the user to click on a link to an infected web page which will compromise the users computer. Internet Service Providers (ISP's) and specialist SPAM filtering services dedicate huge resources to this problem, eradicating around 90% of the problem [191]. Despite this a huge deluge of SPAM messages do get through to the end user inbox, causing an irritant and potentially causing harm. A number of systems have been developed to analyse the content of an email to determine whether the author is human or machine. The majority of these systems borrow from authorship attribution and utilise functionality from a number of the methods we have discussed including Key word detection [13], Support Vector Machines [51], Naive Bayesian [12] and text clustering techniques [158]. It is clear to see that the SPAM filters of today owe a lot to the field of forensic Linguistics.

2.3.2 Twitter

The analysis of twitter messages in an attempt to detect authorship attribution is a very active field at the present [23] [110][174]. With a twitter message you are severely limited to 140 UNICODE UTF-8 characters. Due to this restriction, the number of features that are appreciable for the use in authorship attribution are greatly reduced. There are however the special features that are unique to tweeting of tags (signified by a #, linking a topic) and usernames (signified by an @, Showing the tweet is to that person or in response). These specialist features do assist in authorship attribution, and the removal of them seriously compromises the ability of authorship attribution [110]. Layton also discovered that there is an important threshold of 120 tweets per author at which the capability of authorship attribution of another tweet becomes possible [110]. The practice of retweeting (resending an original message with your identity attached, but acknowledging the original tweeter) add further ambiguity into the mix[23]. It is often not clear if through retweeting the author is simply agreeing with the statement or claiming a statement of their own.

Additionally from an authorship perspective, should someone choose to retweet the retweet, the link to the original author is lost altogether. Twitter messages have now been used to convict, so it sill only be a matter of time before the question of authorship attribution arises and is tested in court. This is important work for the purposes of the research. It can be applied to a situation when analysing a series of packets, or as individual packets it may be possible by only looking at a small sample sizes.

2.3.3 Source Code Analysis

Source code analysis is the processing of a computer program. A computer program can be in one of two states. Firstly a programmer writes the code in a human friendly form which is known as the source code. Secondly a process of compilation is applied to the source code which converts it into a computer friendly form known as a binary. In some cases (depending on the language and compiler used) it is possible to reverse the process and revert back to the source code. The primary focus for Source Code analysis is for the detection of plagiarism in academia and the detection of Intellectual Property theft from commercially sensitive programs [104]. Hayes has performed an in depth analysis of source code analysis and determined "Due to the restrictions in programming language compared to language, a programmer is likely to be more consistent, but there is less variety to detect"[73]. Despite this there have been success in the field when Frantzeskou successfully attributed authors in a small closed corpus through the use of N-Gram analyses at the byte level [61] [62]. This proved that in a limited corpus it is possible to perform successful authorship attribution. My concern with Frantzeskou's methods is that they appear far to simplistic and potentially fall into a number of the pitfalls highlighted by Rudman and explained later in this report.

2.3.4 Music

There are a number of appreciable features in music which are suitable for the types of authorship analysis performed on text [186] [92]. Whilst careful consideration has to be given to the genres, musical styles and the use of different instruments being used, it is possible to perform a qualitative analysis to detect authorship. In fact, dependant on your choice of analysis, the musical instruments used can be a very good author feature [16]. Kevsel [96] however suggest another approach. A modified n-gram analysis of the music which had been prepared for processing would provide accurate results towards authorship attribution. There is also the entire sub-field of forensic voice, who use signals processing techniques to distinguish human voices, which would be equally valid in the analysis of recorded music.

2.3.5 Critical Linguistics

Critical Linguistics is the study of language used in a document in an attempt to assess social and political information that an author may have [60]. The theory is that your choice in words and structure are based on your own ideology. This is very much a qualitative study of a document. There has been a great amount of criticism [208] around Critical Linguists, Stubbs [179] for example, states that "A repeated criticism is that the textual interpretations of critical linguists are politically motivated, and that analysts find what they expect to find, whether absences or presences". Due to the qualitative nature of this analysis it would not be suitable for this research and an artificial language is unlikely to have a political opinion!

2.3.6 Art

The attribution of works of art shares a number of qualities with Forensic Linguistics [56]. There are two primary problem domains with the artwork field. Firstly there is the question of forgery detection. When a painting is to be sold the authenticity of that piece of work must be verified to enable the sale. Because of the vast sums of money that exchange hands when artwork is sold it has been the target of many criminals [55] [24] resulting in some very elaborate attempts to clone or duplicate works of art. The second problem is around the discovery of a new piece of artwork and attempting to verify the original author. In both of these examples the experts are picking a number of features in order to compare and contrast to other known pieces of artwork by that author. The choice of feature set is at the selection of the art specialist who will choose the subset that they feel will clearly distinguish an individual painter based on that painters own style. Arguments can therefore break out when two experts choose a differing set of attributes and come to differing conclusions. Due to forgery cases being a matter of law, the methods used in such a situation must be able to hold up to the rigours of court. More recently artificial learning techniques have been used in an attempt to improve the current capabilities for forgery detection [119] [143] [114]. These techniques have considered both the feature selection problem as well as the analysis using very similar techniques to those discussed for forensic linguistic analysis earlier in this paper. Because of the similarity to previously discussed techniques this area was not considered any further.

2.4 Issues with Forensic Linguistics

There are a great number of acknowledged issues with the practices of modern linguistics which have been aired very publicly [72] [26] [151] [153] and should not be ignored when considering the techniques. A number of the techniques that are applied are contradictory with each other and yet all claim to be based on scientific development. Rudman is the fiercest critic of modern linguistic techniques [151] and has

highlighted six regular pitfalls of modern forensic linguistic practice with regards to “cherry picking”[153].

1. Selecting Primary Text
2. Selecting Quantifiable Markers
3. Selecting Statistical Tests
4. Selecting Control Sets
5. Stopping the Analysis
6. Ex Post Facto Analysis

2.4.1 Rudman’s Pitfalls

Rudman raises some quite legitimate issues which should be considered both in the analysis of articles read and in the development of any experiments and techniques.

2.4.1.1 Selecting Primary Texts

The selection of the text used for analysis is critical. It doesn’t matter how good the technique that you try to use is if your original dataset is flawed. Within linguistics Rudman suggest that a number of academics cherry-pick their original sources which can give a skewed or biased result. This cherry picking is not necessarily an active step to mislead the reader, although this may be true in some cases, but rather academics picking sources that are readily available but not necessarily correct for the job in hand. Examples include choosing a version of the text that is already in a machine readable format, using collaborative texts with multiple authors, mixing genres and/or time periods, using authors from different countries where a number of social differences may impact writing style. The version of a text is vitally important. Selecting a version simply because it is the easiest to obtain or is in the correct format is not a valid strategy. As an example Hargevik [154] studied the works of Daniel Defoe and found that the word “further” appeared only twice in the original publication, but appeared up to twenty seven times in later editions of the works. If you had chosen this word as an indicator of authorship you would arrive at two completely separate results simply due to the version. Selecting texts that appear in publications is always dangerous as you are unaware of the editing that has been performed by the editor and not by the author. The fact that there has been any editing at all could be a problem, depending how heavily the text has been edited and by whom. This is especially relevant to computerised creation of text where the auto-correct functionality, computer based dictionaries and auto-complete functionality all attempt to aid the author but ultimately compromise their text. It may be the case that the program used for the transmission will format

the packets in a specific way, so it is important to be able to detect what a machine has done in comparison to what an author has done. The selection of corpus is critical [20]. The selection of texts from multiple genres or time periods can lead to specialist or period dependant language being used in those texts and not others. In fact this is one of the key indicators as a detection method (See Section 3.5.1). It is important to be able to distinguish between an authors choice of language and genre specific language that has had to be used because of the subject of the text. Within an artificial language this is likely to be defined by the protocol in use. There would be little point in comparing two differing protocols for the content (although some attributes could be tested), and differing versions of the same protocol may have substantial differences.

2.4.1.2 Selecting Quantifiable Style Markers

The selection of style markers is one of the primary questions of Forensic Linguistics. Rudman argues that a number of academics select a style marker, not because of its validity, but simply because of its ease of detection and comparability. It is not practicable to assess the many thousands of style markers that are available for a text, however there must be some methodology behind the choice of style markers beyond "these ones worked for this author". This approach could lead to style markers being chosen that are only coincidentally suitable given the test corpus, and are not in fact an indicator of the author. It is also noted that a number of academics will take readings on other style markers which disprove the author of a text and are therefore ignored in the final reporting of findings. Selecting only the results that back your hypothesis is not good scientific practice. This issue is recognised and there is currently no agreed way of selecting the style markers that you are to use for attribution purposes. To further complicate matters different markers will be relevant for different forms of media, and are not necessarily author specific markers.

2.4.1.3 Selecting Statistical Tests

With the selection of a statistical approach there will be a number of assumptions that are made, such as the distribution of unique words is equal throughout the text. These assumptions, although potentially wrong, should be understood by the authors as to the effects on the statistical method used. Some of the greatest controversies in Forensic Linguistics have arisen through the misunderstanding of the significance of the statistical functions being performed. The most famous example of this was the CUSUM debacle [80] [70] [76]. The CUSUM (Cumulative Summing) method [128] of statistical analysis has been used for the detection of authorship attribution, and was used within the court system and a valid method. The method centred around the attributes of individual sentences, assuming that the habits of an individual author with regards to sentence length and word length will be consistent for every text they write. Not only had the authors made bad assumptions, they didn't appreciate the statistical method that was being used and had no real idea of how it was working. They likened their discovery to that of fingerprints analysis [54] when, for

the first 50 years of its use in court, the science was not fully understood but was accepted. This method came into serious question within the academic world and then ultimately failed during a demonstration live on television when a convicted felon and the chief of justice, England were compared using the technique and found to be the same author. The method has now conclusively been proven not to work [29] and is no longer used as a forensic Linguistic technique within the United Kingdom [43]. Whilst CUSUM is the most extreme example of the misuse of statistical analysis, this is not the only example within Forensic Linguistic practices. What is clear is that if a statistical method is chosen, that the implications are understood and the variance and error should be calculable and displayed with any results that are shown.

2.4.1.4 Selecting Control Sets

The selection of control sets has many of the same issues as the selection of primary texts. Any control set should be as close to the primary texts as possible to avoid the possibility of the content causing the variation rather than an author. Care should be given to the subject matter, date of publication, Controls with multiple authors and the use of authors of another nationality. Additionally there are a number of studies where the test case is of a single piece of text [161]. Clearly this can not be used as showing statistical value. The size of the control set should be large enough to be able to deduce statistical reliability of comparable or larger size than the primary texts. The samples must be representative, of sufficient size and be completely random.

2.4.1.5 Stopping Analysis

There have been examples of Forensic Linguists stopping their analysis prematurely because the initial answer that is received prior to the completion of all tests gives the results that they were expecting [59]. This is despite the possibility of a better fit potentially being available within the corpus. Rudman compares this [152] to a case whereby 6 DNA loci were used to prosecute an individual, however the conviction was overturned when the test was performed with 10 DNA loci, and the additional 4 were found not to match. If an experiment has been clearly laid out it should be followed to its conclusion.

2.4.1.6 Ex post Facto Analysis

This is the case when any of the above are manipulated after one set of results in order to produce a more desired set of results. This is almost impossible to detect unless it has been stated that manipulation has occurred in the writing up. Normally this takes the form of troublesome texts being removed from the corpus [79] or the text being manipulated in order to remove troublesome features [141]. This practice is not necessarily a bad thing as long as the purpose of the exclusions is fully explained and the same scientific reasoning is applied to the texts that are not being removed from the experiment.

2.5 Cyber Attribution

Currently only Forensic Linguistics approaches that may be suitable to solve the problem of Attribution in a cyber context have been considered. To fully appreciate the relevance of these techniques the current approaches being employed to attribution in the cyber domain must be considered. The seminal text in this area is by Wheeler [196], who splits the attribution techniques into seventeen technique styles. These are ;

1. Store Logs and Traceback Queries
2. Perform Input Debugging
3. Modify Transmitted Messages
4. Transmit Separate Message
5. Reconfigure and Observe Network
6. Query Hosts
7. Insert Host Monitoring Functions
8. Match Streams
9. Exploit/Force Attacker Self-Identification
10. Observe Honeypots/Honeynets
11. Employ Forward Intrusion Detection Systems
12. Perform Filtering
13. Implement Spoof Prevention
14. Secure Hosts/Routers
15. Surveil Attacker
16. Exploit Reverse Flow
17. Combined Techniques

This is a useful taxonomy and has been accepted in many papers [134] [18] [40]. The taxonomy will be used to explain each technique, consider the advantages and disadvantages of each method, and expand upon the latest developments in each field.

2.5.1 Store Logs and Traceback Queries

This method involves the storing of logging information on routers which would provide information as to where a packet had come from, where it was going and what the content of that package was. There would be some method of querying a router to ask if it had seen a specific packet before and where did it come from. With this method you can continually probe the router at the last known location and traceback to the originator. In slight variation on this, the router simply stores the contents of the packet and is then probed to see if it has seen that packet before, without storing where it has come from [196].

The advantage of this technique is that is could enable complete traceback capability as well as the capability to perform historic traceback for as long as the log information was stored. In this scenario it would be possible therefore to detect an attack post event, but still use traceback to provide the source of the attack once the attack has been identified. This solution quickly runs into scalability issues [201] and so there have been major efforts to consider the minimum amount of information required to store in order to enable a full traceback [201] [47]. Examples include only storing that hash of a message rather than the entire message [171] [172]. Further issues are experienced when you consider that the router upstream may not be under your control, but under the influence of an adversary and as such could not be relied upon [18].

This is a very active area of academic research, with new versions of logging schemes being developed [201] improving on the amount of information that would have to be stored to enable a successfully traceback. Fundamentally they all fail to scale to the current size of networks and network traffic and ultimately would require a fundamental change in the way in which the internet currently works.

2.5.2 Perform Input Debugging

This method is based around developing a signature for an attack. It is regularly employed in Network based Intrusion Detection Systems (NIDS). An attack is recognised and a signature is developed for that specific attack. The signature is then pushed out upstream to you NIDS. Should the attack happen again then the NIDS will instantly recognise the attack and flag to an administrator that another attack is in progress. Because you now know which routers the attack has passed through you are able work out the route at least to your perimeter. In order for this method to enable complete traceback capability these signatures would have to be pushed out to the Internets backbone routers and potentially an adversaries network. This would increase the processing burden on the backbone routers and could potentially be used as a form of attack in itself to push many signatures to a router and overwhelm the network. It would provide an adversary clear intelligence that their attack had been detects when you push a signature onto their system. Developments in Network Based Intrusion Detection systems have been primarily focused on anomaly detection [111] [130] and improving detection capabilities (both increasing positive detections

and removing false positives) [135] [105].

2.5.3 Modify Transmitted Messages

This method involves modifying a packet as it travels from node to node. In the most simplistic version of this process, each router that a packet passes through would append to the packet a unique stamp for that router. This means that when a packet arrives at the destination it would have the complete route appended to it. For a route with a large number of hops this method can significantly increase the amount of data being transmitted, as each node on the route adds its own data [138]. The current version of the IPSEC packet is explicitly designed with this capability [103] [15], although unfortunately the take up of this protocol has been slow to get established [86] With the limitations of marking every packet at every node, attempts have been made to provide similar levels of traceback capability whilst only modifying a random number of packets. This area of research is referred to as Probabilistic Packet Marking (PPM) [138]. PPM only works reliably when a number of messages are received as a part of an attack [67]. You are also reliant on the routers that you are passing through to provide accurate information. The thrust of academic research in this area is to improve the efficiency of packet marking schemes [8] [116] [7] to get the correct balance between the overhead of marking packets with the capability of traceback combined with ever more efficient methods of marking the packets through using unused sections of the packet, thus not increasing the packet size [6].

2.5.4 Transmit Separate Message

This method is very similar to modifying transmitted messages, with the exception that instead of the additional information being appended to the packet as it traverses a router, an entirely new packet is created and sent to the destination address. The process in fact creates even more data than modifying a transmitted massage as each packet has the overhead of a packet header. One massive advantage of this approach is that the separate message can use a different route from the original packet. In fact some designs even have an entirely separate network to perform this function [178], thus leaving the primary network free for actual data you wish to transmit. There is naturally an inherent cost with running a second network which means that the take up of this solutions is highly unlikely. One method that has gained traction in this area is that of iTrace. This is a modified ICMP traceback packet developed by the Internet Engineering Task Force (IETF) which has shown good results [19] [9] [121]. More recently there have been attempts to enhance the capability of iTrace to cope with multi-path attacks [31].

2.5.5 Reconfigure and Observe Network

This method involves detecting an attack and monitoring those connections for changes as you modify the network. The most simplistic version of this would be to increase the TTL (Time To Live) of each packet by a different amount on each router. So if you have three potential routes to the machine that is under attack you increase the TTL on router A by 3, router B by 5 and router C by 7. You then monitor the packets from the attack to see how much the TTL has increased, thus giving you the answer as to which router the packet is passing through. By using prime numbers you will also be able to work out if the packet is travelling through two routers, and which ones they are. Other changes that you can make to the network include modifying the network topology (modifying the routing tables) and changing firewall rules. Unfortunately any changes that you do make to the network must be seriously considered as there is a serious possibility of causing the network to fail, or having undesired effects on your own infrastructure. These methods are only suitable during the time of an attack and you are limited in only being able to adjust the configurations of the routers that you control.

One method that has been proposed which could be extended beyond the network that you control is ”controlled flooding” [30] [160]. Essentially this is performing small scale denial of service attacks against external infrastructure to see if it has an affect on the attack. If the attack slows down during the controlled flooding then it can be assumed that the router that you are flooding is a part of the route back to the attacker. This method has dubious legal implications that are beyond the scope of this review.

2.5.6 Query Hosts

Through the use of tools that are pre-installed on a host system that has been compromised it might be possible to extract information about an attacker. This would simply be a case of analysing logs,using in built tools such as netstat and performing a forensic analysis of the host. It should be noted however that if the host has been compromised, the integrity of both the services and logs should be brought into question as it is possible for an attacker to modify logs and services in an effort to cover their tracks.

2.5.7 Insert Host Monitoring Functions

Beyond simply using the pre-installed services to perform your host analysis, it is possible to install third party services that perform similar and additional functions. A great number of these are commercial products and as such their internal workings are commercially sensitive and not available in open literature. This functionality is usually built into a larger security product such as an IDS (Intrusion Detection System). Network intrusion detection systems can be used to deploy agents on individual hosts which report back to a central server. Should a breach be detected, the agents can relay this information to a central server

which can then advise as to how to reconfigure the network to counter this threat. As a result any attribution information is extremely valuable, as it can be used to mitigate against an attack closer to the source. This method has the advantage that it is less likely to be compromised by an attacker, as each developer will have their own host monitoring processes, requiring the attacker to target the specific process that has been inserted rather than a standard process installed within the operating system. The disadvantage however is that the process is running on a compromised system and so can not be fully relied upon as the attacker may compromise the process to give false information, or fail completely. Additionally if the process is added to the compromised host post attack, the attacker may detect this and realise that they have been detected.

One very controversial area that the insertion of host monitoring functions can be used is in the process of ”hack back” [87]. This is the process of detecting the stepping stone from which an attack is being performed and then breaking into that stepping stone yourself in order to trace back to the next step in the chain. It is included under the heading ”insert host monitoring functions” as this is the next step once you had compromised the stepping stone machine. There are very serious questions over the legalities of such techniques. In the worst case scenario you could be hacking into the incorrect system, if other traceback techniques have failed, but more likely you will be compromising an innocent victim of malware who is unaware of the attack. This is an area that continues to create great debate [125] [65] [25] and is unlikely to be formally accepted any time soon. As a result there is little in the public domain about the techniques [89].

More recently, with the greater adoption of virtual machines, introspection has become an expanding area of research [115] [133]. With more modern powerful computers it is now possible to run multiple versions of an operating system on a single host, with an overarching host operating system. For example a physical box may run an operating system of ESXI which is specifically designed to run virtual machines. You could then create a number of virtual machines running various versions of Windows and GNU/Linux. These virtual machines would run entirely independently of one another and would not be able to interact with each other. It is however possible from the host operating system (in this case ESXI) to both interact with, or passively monitor the virtual machines operating system. The process known as introspection is the passive monitoring of processes running live on a virtual machine from the host operating system. This can be used for the analysis of an ongoing attack, to gather attribution information as well as learn about the adversaries techniques. This is an improvement on using the compromised operating system, as the host operating system will not be circumvented by rootkits.

2.5.8 Match Streams

This process involves the careful analysis of all messages entering and exiting a network or a host. Through careful analysis it may be possible to match the streams entering the network location with those exiting

the location without having to interact with the host itself. A number of approaches to match the streams can be used including the analysis of message headers [203] [22], the analysis of the data content [32] or the careful analysis of the timing of a stream entering and exiting a node [180]. All of the methods rely on some form of link analysis for which a number of techniques exist [132], even if the algorithms have not be specifically tested on this use case. The advantage of this system is that you are not reliant on a host which may be compromised or not within your control to perform the analysis, you simply need access to the network in which the host resides. Unfortunately the methods are reliant on accurate timing data, and very small discrepancies can prove fatal. Additionally this is not a method that scales particularly well, and as networks become more complex the analysis of the streams becomes ever more difficult [98]. The most recent research in this area has been focused on increasing the speed and bandwidth in which current methods can work [180] as well as expanding the methods into radio and mobile based networks [58]. These methods are more akin to Electronic Warfare and will not be expanded upon within this paper.

2.5.9 Exploit/Force Attacker Self-Identification

This is the process of carefully analysing what the attacker is doing in the hope that this will lead to key indications as to their identity. There are a large number of ways in which an attacker may provide identifying information. The Forensic Linguistic techniques discussed earlier in this paper provide one example of how the attacker could self identify themselves. These methods were covered extensively earlier and will not be expanded upon here. Other ways in which an attacker may give away information include;

1. Data included in the attack. For example a spear-phishing email is likely to include a return address, links to an external server and/or an attachment, all of which will be specific to that attacker.

2. Self Identifying protocols. Files included in the attack could include identification marks. For example Microsoft Office embedded the MAC address of the original authors PC in documents created using Microsoft Office 1997 [156]

3. Beacon Functions. This is a tool with a call back function. As an example an email could be sent the attackers spear-phishing account. Included in the email could be an embedded HTML picture link. Should the email be opened with HTML enabled, the HTML link will automatically download the image by contacting the server it is hosted on. This could result in a direct connection to the attackers IP address resulting in him leaking this crucial information. Further to this it would be possible to create a unique HTML link for each attack. A similar approach can be used on an HTML web page.

4. Cookies. Through providing simple text based cookies, should an attacker fail to clear their cookies after an attack it would be possible to associate them to their previous visit [209]. Further to this it is

possible to extract data from other stored cookies created by other sites [185]

5. Tabs. Should an attacker use a modern browser with multiple tabs open it is sometimes possible to extract information from the other tabs that are open through various browser extensions [124] [74].

6. Theft Tracking Tools. There are a number of tools that are loaded into the BIOS/UEFI which are designed to provide traceback capability on the theft of a device. These are designed to work even after the operating system has been re-installed [44].

7. Watermarking. This method involves using stenographic techniques to embed hidden watermarks into your files [187]. Whilst this will not help in immediate capture of an attacker, should an attacker be caught through other means, the watermarked files will enable the quick identification to the attack. This is especially true if the watermark can be made unique to the attacker [88].

Unfortunately not all of these methods are appropriate for every type of attack. Additionally once you are aware of the methods they are very easy to mitigate against, and so would present little threat to a skilled attacker. As a result the focus in this area should be the creation of new techniques.

2.5.10 Observe Honeypots/Honeynets

Honeypots are hosts systems that are designed to look enticing to an attacker whilst no containing any actual data or processes [173]. It is hoped that through running a honeypot the attacker will perform their attack again the honeypot rather then against a live system, thus not affecting running of the business in any way. The honeypot can also be used to gather information about the attacker such as the tools and techniques that they are employing. This information can then be used to help protect the rest of the network. Honeypots are graded on how interactive they are. A low interaction honeypot will only provide a small number of simulated services which will only provide a very small number of pre-programmed responses. As a result you only get a very limited view of the capabilities of an attacker with a low interaction honeypot. However a high interaction honeypot is essentially a fully working system and will provide a much greater resolution as to the hackers activities. Honeynets are entire networks of honeypot machines, or a completely vitalised network designed to imitate entire sections of a network, or even a subnet[148]. The ultimate goal of a honeynet is the same as a honeypot, to provide intelligence on an attacker. Through providing multiple machines on a honeynet you may be able to glean additional information such as the type of machine the the attacker is going after. For example, do they attempt to attack the Domain Servers to gain control of the entire network, or do they attack the FTP server, in which case the attacker is likely to be after a file. Tarpits are another form of honeypot/honeynet [146]. Rather than gather intelligence these are specifically designed to use up the resources of an attacker. These are especially effective against automated tools such

as password brute forcers and worms. Honeypots/Honeynets are extremely good at providing intelligence about an attacker, providing they are enticing enough that the attacker actually goes after the honeypot. The disadvantages of honeypots is that they can be very resource intensive to maintain from an administration point of view, especially for a high interaction honeypot/honeynet. There is always the fear that these vulnerable systems could be used as a stepping stone for another attack, and so may be used against you. Recent developments in Honeypots/Honeynets expand the area into more specific forms of network such as ICS and SCADA control systems [189] to reflect the current threats.

2.5.11 Employ Forward Intrusion Detection Systems

This method involves placing Intrusion Detection Systems as close to potential attack avenues as possible. Ordinarily Intrusion Detection Systems would be placed within an internal network and used to detect an attack happening. The reasoning behind placing them as far upstream as possible is that they provide much better traceback capability as they are closer to that attacker. This would be suitable for a company with a large digital estate, or as a nation state capability, but would be less effective for a small company with no external penetration. With this method you are reliant upon the IDS being able to detect the specific attack and that the attack is routed through your IDS. Recent developments in this area have included a number frameworks for a network forensics system [194] [202]. It has also been proposed that the system could be modified to also detect information egress from a data breach or attack [117].

2.5.12 Perform Filtering

This method involves setting strict rules at your network gateways that block packets that do not include enough information to enable some form of traceback. This should mean that only packets that have a capability of being traced will enter the network [159]. Therefore any attack that is not traceable will automatically be dropped at the gateway, and if it is traceable the attack will be let through and hopefully detected by Intrusion Detection Systems. This should be easy to set up as the protocols are clearly defined. Unfortunately the trace may only be to the last stepping stone, but if the is an immediate trace to that point then you can hopefully work with that stepping stone to stop the attack. It may also be possible to use Advanced Evasion Techniques to bypass the filters.

2.5.13 Implement Spoof Prevention

In an attempt to avoid Advanced Evasion Techniques, spoof prevention should be implemented to detect manipulated packets. Spoof prevention is specifically looking at the packets as they arrive to detect that the packets are correctly formatted and have not been manipulated. This is subtly different to ingress filtering

which is focused on the packets containing enough information to provide traceback, and not their validity as a packet. The recommendation is to use the gateway to block manipulated packets, and for protocols that are easily manipulated but difficult to detect, provide an encrypted VPN tunnel. Correctly formatted packets are a lot easier to traceback and make ingress filtering more effective. Developments in this area have primarily focused on improvements in techniques for spoof detection [164] [190] and the extension to new protocols [144]

2.5.14 Secure Hosts/Routers

This is the process of securing as many hosts and routers on your network as possible. The theory is that if there are fewer vulnerable hosts on the network then there are fewer potential avenues for attack, thus reducing the number of hosts that require analysis, making traceback quicker [196]. Additionally host should be configured to enable the connection to more secure services, which provide an element of traceback. There is no specific method of traceback considered in this section.

2.5.15 Surveil Attacker

If potential attackers have been identified then it is sometimes possible to surveil them in order to learn more about them and how they operate. An example of this would be the Anonymous collective. These are a large number of individuals who will sign up to specific ”ops” which they feel are in line with their own beliefs. The individuals will then meet in IRC chat rooms to discuss how to proceed with an operation and who to target. Each member will have a nickname when accessing the chat room and it may be possible to build up a profile of that attacker based on their profile name. When an attack is successful the attacker likes to claim the attack as their own and will often use their nickname against the attack. They may also own twitter accounts under the same nick name. Through surveiling the IRC chat rooms and twitter account you may be able to extract information about the actual person behind the nick name.

2.5.16 Exploit Reverse Flow

The majority of transmissions across the internet are bi-directional. When an attacker communicates with a host there will be some form of communication back. This method involves manipulating the returning communication in such a way that it can be used against the attacker. One approach is to embed your own watermark in the return communication and use your IDS to detect the exit point on the network [193] [206]. The latest developments in this area have extended the techniques to wireless technologies [192].

2.5.17 Combined Techniques

Combined techniques are essentially any of the previous techniques used in combination with each other. This could include two or more techniques. These tend to be used in commercial packages rather than within research.

2.6 Conclusions

The literature around attribution has been thoroughly explored. Starting with an understanding of historical linguistic techniques it became evident that there were a large number of branches of linguistics however not all of these are going to be suitable within a cyber context. The techniques employed simply cannot be transferred to the cyber realm. When switching focus to the purely linguistic attribution it was discovered that the literature did expand in to a range of different techniques that would be transferable into the cyber realm. There were a number of significant questions over the academic practices used in these early works. Suspicious practices included specifically choosing the primary texts, attributes, statistical methods and control sets in order to get a publishable result as well as stopping analysis part way through once the preffered result has been achieved and devising the hypothesis after the analysis has been completed. These findings significantly damaged the field of linguistics for a considerable time and whilst the reputation of the field was being rebuilt, the outputs dried up. Since then the field has had a resurgence and is once again a reputable academic field under the banner of ”modern linguistics”. Unfortunately the damage was done and the modern approach is attribute analysis on large data sets, creating of data marking points and the creation of algorithms for the processing of the data. This is an approach that has existed in the computer science field for a considerable amount of time and the linguistic filed has more to learn from computer science that the other way around. As a result although a number of the attributes highlighted in the literature are of use, the general techniques have already been applied in a cyber context.

Having considered the current techniques for traceback attribution based upon the framework laid down in [196]. Each of the methods proposed has been analysed, considered the advantages and disadvantages of each method, before exploring the latest developments in each method. From this analysis it was clear to see that Forensic Linguistic attribution techniques falls within ”attacker self identification”. There were a number of key points discovered whilst exploring the literature:

- A great number of the methods presented in academic literature appear to solve the problem of tracing back a Distributed Denial Of Service (DDOS) attack and as such their focus is on large scale attacks, with huge numbers of packets being traced back during the attack. Whilst DDOS does still remain a threat, the numbers of these types of attacks are extremely small in comparison to more

traditional hacking or the more recent trends towards ”Advanced Persistent Threats” (APT) [131]. This is unfortunate as it appears to be the primary focus of academic literature.

- A number of these technical solutions require a fundamental change to the way in which networking and the internet works. Whilst some of these changes may be implemented in the future, there will be a considerable amount of time before all elements of the internet become compliant, if at all. Additionally there is the assumption that all segments of the internet will play by the same rules. This is a massive assumption given the offensive nature of some of the activities on the internet which are originating from states, whom have complete control over their countries infrastructure. You would not expect an attacking state to adhere to the rules which would trace an attack back to itself. These are clearly not viable solutions for the immediate problem.

- There has been a trend in more recent literature to extend the attribution techniques to wireless media. This is consistent with the development of technologies moving towards RF enabled data networks. The initial studies in this area show that there are distinctions as a result of the underlying protocols used to control the radio’s [58]. The trend for RF based research is likely to continue as the wireless provision continues to improve and is more widely exploited.

- It should also be noted that there are a number of commercial offerings in this area’s whose techniques are not in open literature.

It is clear from the literature that there are still a number of issues to overcome with the attribution problem and that research is still active in a number of areas.

Chapter 3

Framework Development

3.1 Introduction

This chapter will document the design process in developing the proposed cyber attribution and threat actor frameworks. There will be discussion around the design choices and ultimately an explanation of why specific decisions were made over competing options.

3.2 Design Requirements

In order for the framework to succeed in its intended goals it must meet the requirements defined below

- The framework must be mathematically based in order to enable validation of the techniques used
- The framework should highlight intelligence gaps and suggest methods of filling those gaps
- The libraries should be complete for a specified subject area, highlighting attributes which could be considered
- The framework should enable specialists from different areas to work together to create a more complete attribution picture

3.3 Actor

An *Actor*, A, is an entity or person who is performing an action through cyber means. In its simplest form an actor can be be thought of a series attributes which belong to that actor. Through establishing all of the attributes that belong to an actor, or at least establishing enough attributes to uniquely identify the actor, you will have been able to establish an identity for that actor. This identity should directly link to an individual.

A=$\{Attributes\}$

3.4 Attributes

An *attribute* constitutes a discrete and identifiable feature of an actor. It must be both measurable and identifiable. Each attribute will be in the form of a tuple, as each attribute will have an associated confidence value. The confidence value should refer to the confidence that the attribute is inherently owned by the actor, rather than the accuracy of the attribute. For example, in the case of an IP address, it is trivial to spoof an IP address when performing actions such as a denial of service attack. In a situation like this the IP address is highly likely to be randomly assigned and bear no relation to the attack. If however the IP address in question is being used by malware to beacon back to it is almost certain that the IP address is associated to the actor and therefore is an attribute of that actor. It is almost certain that IP address being used to beacon to is not the actors own IP address, but it is still an attribute of the actor, meaning in this case the confidence value would be high.

The confidence value itself can be represented by any numerical value, as long as all analysts agree on a standard prior to using the framework. A value of 1-5 (1 being low confidence, 5 being certain), a percentage of certainty, or a number relating to a position on the uncertainty yardstick would all be suitable. For the purposes of testing the framework the values 0-1 have been chosen, with 0.2 intervals, with 0 representing highly unlikely and 1 representing almost certain. The default value will be 1 unless modified by the analyst. This will form the mathematical basis of the confidence value.

$$\text{A}=\{Attribute1, Attribute1_{Confidence}$$
$$Attribute2, Attribute2_{Confidence}, \dots\}$$

3.4.1 Classes of Attribute

In its current format the attributes assigned to an actor can have any value of any type, whereas the confidence value is formally defined. For the benefit of the model a number of classes of attribute must be defined in order to allow comparative operations to be performed and assist in the analysis.

The defined classes are:

- Text_String
- True_False
- Integer
- Real_Numbers
- IPv4

- IPv6

Class Text_String is defined as any block of alphanumerics. Examples would include an E-Mail signature block, PGP key or a User agent ID. A string comparison would enable you to perform difference analysis.

Class True_False is defined as a single bit that can be enabled or disabled, in the form of a flag OR the output of a truth table. Valid outputs would be True (1) or False (0).

Class_True_False=$\{Attribute1, Attribute1_{Confidence}, Attribute1_{TrueORFalse}\}$

Class Integer is defined as any positive or negative whole number, including 0. Examples would include specialist character counts, W value and port numbers. It is possible that the integer class may have a range of valid values. As an example, several texts by the actor may have been analysed by a linguistic specialist, and a specific W value may have been defined, however a certain amount of range may be preferable. In this situation a class integer would have the additional attributes of maximum and minimum values.

Class_Integer=$\{Attribute1, Attribute1_{Confidence}, Attribute1_{Minimum}, Attribute1_{Maximum}\}$

Suitable maximum and minimum values would be defined by the expert and would enable comparative analysis without requiring an unreasonable amount of accuracy in the function being performed.

Analysis operations would include equals, within the range of, or above the specified value

Class Real_Number is defined as any number that is expressed with a decimal element. Examples may include version numbers, Words per sentence values and frequencies. Once again there may be a requirement for an expert to set a range in order to perform realistic analysis.

Class_Real_Number=$\{Attribute1, Attribute1_{Confidence}, Attribute1_{Minimum}, Attribute1_{Maximum}\}$

Analysis operations would include equals, within the range of, or above the specified value

Class IPv4 is defined as an IP version 4 valid IP address. This is 4 blocks of integer values between 0 and 255. Ordinarily a single IP address would be associated to and action, and therefore an actor, however there may be occasion when an IP range is being used. On this occasion the IPv4 class would have a netmask associated which would define the range of IP's.

Class_IPv4=$\{Attribute1, Attribute1_{Confidence}, Attribute1_{Netmask}\}$

Analysis operations would include Equals or within the range of.

Class IPv6 constitutes a valid IP version 6 address. This is defined as eight groups of four hexadecimal digits, each representing two octets, totalling 32 hexadecimal characters in total. Once again there may be an IP range and thus an associated netmask.

Class_IPv6=$\{Attribute1, Attribute1_{Confidence}, Attribute1_{Netmask}\}$

Analysis operations would include Equals or within the range of.

It should, with these defined classes, be possible to represent any data which could be used as an attribute, and perform logical and systematic analysis against each attribute.

3.5 Libraries

In the current model we have an actor, A, and a series of attributes which refers to that actor. Each attribute has an attribute of its own, confidence, and depending on the data type may have further attributes. This can now be used to create a long list of known attributes for an actor.

A=$\{Attribute1, Attribute1_{Confidence}$
$Attribute2, Attribute2_{Confidence}, \dots\}$

Whilst a valid way of defining an actor, this methodology does not however assist an analyst in highlighting further attributes which may be considered for analysis but are missing. To resolve this issue a series of libraries of attributes are to be created. Each library will refer to a specialist area of knowledge. This area of knowledge should relate to a specific domain of knowledge and is likely to be developed by a specialist expert in this area. This will then create a set of attributes relating to that domain. The advantages of being domain specific is that they can be independent of tools, in fact multiple tools may be used to create a fuller picture, and use multiple different theories as input (for example Yules K measure and Brunet's W measure are both competing hypotheses in linguistics but would both provide a valid input into the model.) Further advantages come from the fact that as tools and techniques develop and improve, the input to the model will improve and thus improve the overall output without modification of the model or analysis technique.

Initial proposed libraries are:

- Linguistic
- Operating System
- Executables
- E-Mail
- Browser
- Network

This is not a comprehensive list of all possible libraries, however this will provide enough variety to prove the validity of this model. It is hoped that experts in other knowledge area's develop their own libraries and add to the model to further enrich the attribution picture and develop a truly interdisciplinary capability.

The definition of a library is therefore, a set of attributes which are linked through a knowledge domain. The benefit to the analyst of having a series of libraries is that, without the deep technical knowledge of the domain they are able to populate relevant attributes based on tool inputs and therefore perform analysis and build the attribution picture.

3.6 Framework

At this point we now have a series of independent libraries containing attributes, however there is a lack of outlining framework in order to place each of these libraries together in order to make any further analysis possible. Furthermore there will be a number of attributes that will crossover libraries and thus build a more vivid attribution picture. Finally we have no verification method to test whether or not the libraries themselves are complete, or are in fact missing attributes. In order to resolve these issues an overarching framework is require to both place the libraries in to, as well as the attributes with the libraries. In order to achieve this it is proposed that the OSI 7 layer model as the basis of this framework. The OSI 7 layer model is an attempt to model all elements of networking by splitting it into different layers, each of which performs a different function which builds to create a complete connection from one application to another. There is criticism of the model [165] however it is a comprehensive breakdown of the networking process which, although not perfect, has been widely accepted. Furthermore the OSI model is taught almost ubiquitously to anybody learning about networking, and so would not need any further explanation to an analyst with simple networking knowledge. The OSI model layers are *Physical, Data Link, Network, Transport, Session,*

Presentation, and Application. This will enable the analyst to place the library and attributes in the correct place on the framework. It will also highlight through the distribution of attributes through the model if there are any gaps which is likely to mean that attributes are missing from the library. The OSI model however does not provide complete coverage from a interdisciplinary perspective. This requires more than simply analysis of the networking, and must include all of the human elements. A common addition to the 7 layer model is an eighth layer, the *User*. This would be defined as an individual who is performing actions on the network connected machine. The User will be a single person with all of the attributes of a human. As a result the majority of the linguistic elements would fall under this layer. The User layer, although referring to a single person, may relate to multiple digital identities. Finally we find that actions from an actor are rarely performed as an individual and are in fact part of a larger group. In order to capture this a ninth and final layer needs to be added which is defined as *Social*. The social element will refer to a group or organisation that is acting as a collective to perform an action against a network. This captures both real world groups as well as entirely digital constructs. Attributes that would fall in this layer would include group name/identity etc.. as well as group specific language identified through linguistic analysis. The final result is a 9 layer model which captures every element of the libraries and therefore all attributes.

- Social
- User
- Application
- Presentation
- Session
- Transport
- Network
- Data/Link
- Physical

3.7 Physical/Digital

The model now encompasses every element of a digital existence captured in a simple to use framework. Whilst useful in solving the attribution problem if you limit yourself to purely the digital domain you will never achieve actual attribution to a human operator. This is the ultimate form of attribution and is often

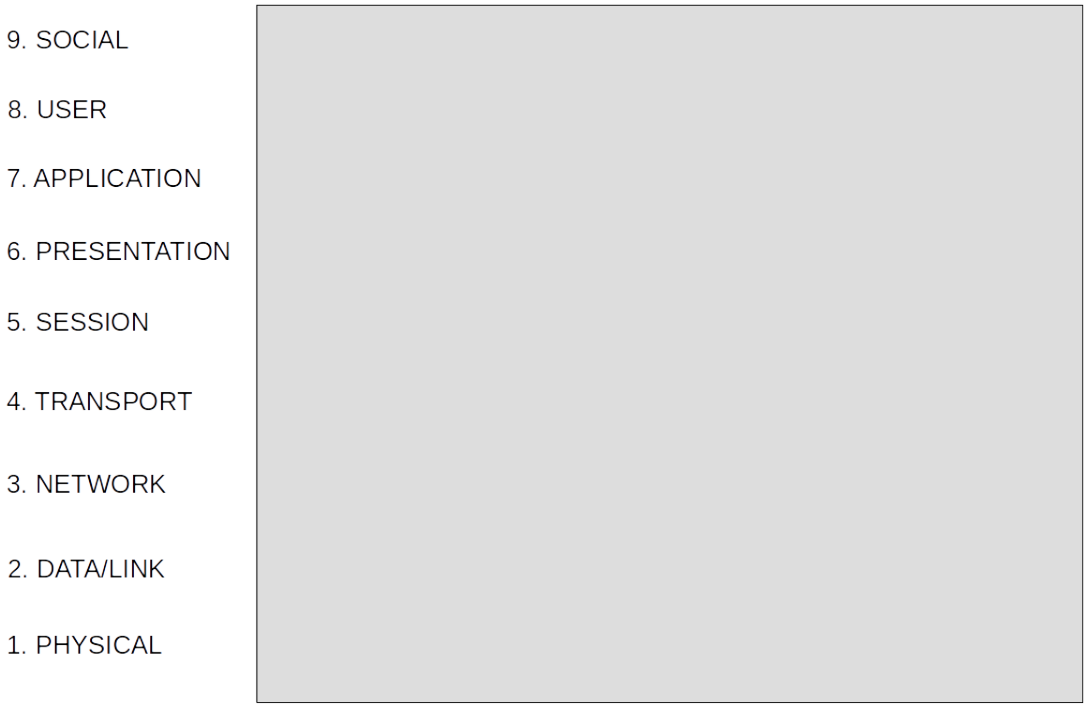

Fig. 3.1. OSI + 2 model

the goal. As a result, to further enhance the model an additional dimension must be added for the physical world. It can be argued for each digital instance a physical device must exist. This means that for every digital attribute listed there is an equivalent real world object attached to that digital existence. In some cases the physical device may simply be a server sat in a data centre which provides no additional benefit, however there is also a real world person sat at a real world computer in a real world location. All of these sit within the physical domain, have physical attributes which may assist in the ultimate identification of an actor. As a result these must be captured by the model. This results in the final framework looking like this:

3.8 Conclusion

This now provides a working framework which can be used by experts to develop libraries. Gaps in the libraries will be highlighted through the distribution of the attributes within the model. The libraries can then be used by analysts to populate the model for each actor to enable the analyst to build a complete attribution picture which can then be used to identify additional actions by the actor and/or work towards identifying the actual identity of the actor. The framework enables co-operation between technical specialist which can only enhance the attribution picture with a cross disciplinary approach. Finally the model is based on sound

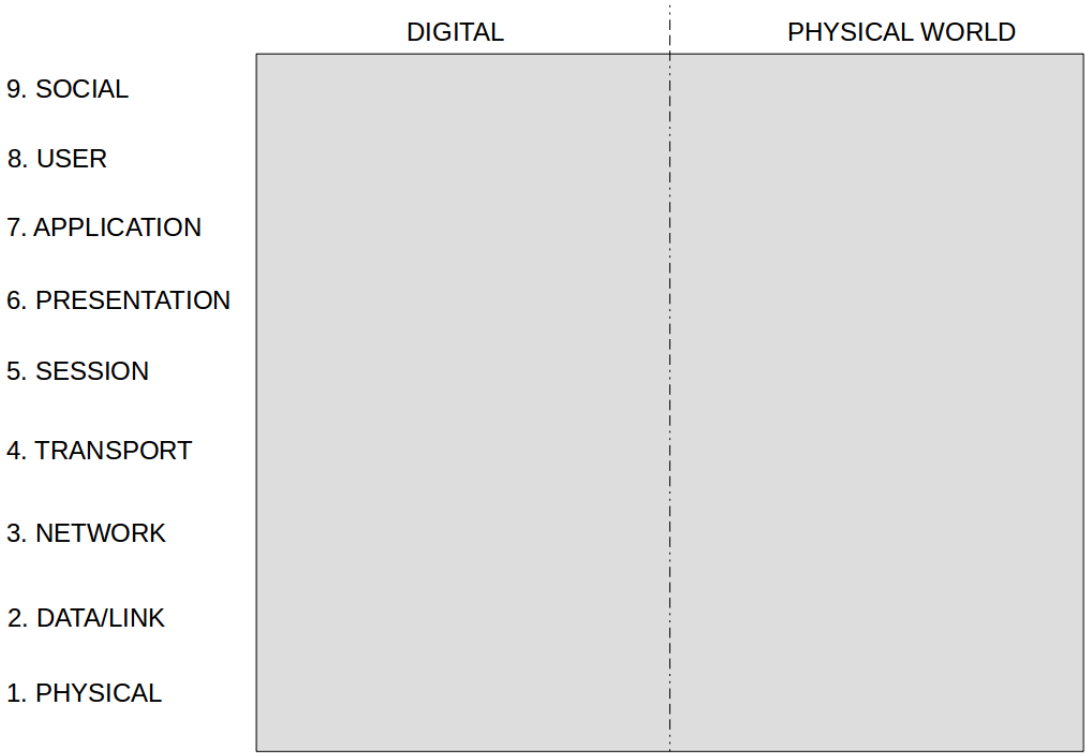

Fig. 3.2. OSI + 2 model, incorporating the physical world.

mathematical principles which may result in the development of new tools an techniques, but fundamentally can be used as an analytical tool.

3.9 Development of a Threat Actor Framework

Cyber attacks against an organisation can happen in a variety of differing ways, with a huge number of threat actors working to perform these attacks. It is impractical to list and develop a profile for every attacker in every instance. Equally there is always the threat of a previously unknown attacker acting against an organisation. It is therefore useful to have a number of archetype threat actors which can be used as an indicative representation of the majority of attackers that an organisation would face.

Firstly we will assess current archetype actors that have been proposed in a number of different forums. Each proposed threat actor will be considered for its merit, and each forum will be assessed as a whole. Once the analysis is complete we will then examine whether or not these threat actor archetypes are used by professionals within the cyber security industry. This will provide an understanding of their requirement within a framework. We will consider a number of different jobs roles and interview persons currently fulfilling these roles. Finally in a proposal of our set of threat actor archetypes and provide specific details

as to the characteristics of each archetype.

For the development of an archetype it must be defined. Collins Dictionary defines an archetype as ”a perfect or typical specimen” [48]. The Oxford English Dictionary states ”a very typical example of a certain person or thing” [49]. Dictionary.com proposes ”the original pattern... from which all things of the same kind are copied or on which they are based” [50]. It is clear that there is not one accepted definition, and that it has differing meaning dependant upon the field under which it is being applied. For the purposes of this report an archetype is a very typical example from which others may be classified. Furthermore to define archetypes of threat actors on cyber systems we must also consider what a threat actor is. For the purpose of this report a threat actor is any actor upon the cyber system that may cause damage or compromise to those cyber systems.

The purpose of having archetypes is to have a simplified generalisation of a class to which certain attributes may be applied. There are many uses for these generalisations. For example Microsoft uses what it considers to be archetype windows users when developing its windows operating system. It can then assess that the needs of each of these user types has been met. When considering cyber threat archetypes, each archetype needs to represent a new element or angle for it to be classed as a new entry. Each archetype should have its own characteristics which make it distinct from other entries. The majority will be attackers, but this may not always be the case. These archetypes of threat actors should then provide a useful function to a number of roles including, but not limited to, risk assessors, Computer Emergency Response Teams, Cyber threat analysts and developers of standard operating procedures.

3.10 Published Threat Actors

There are a number of locations where lists of potential threat actors have been published. The primary source discovered for this documentation was found to be risk assessment accreditation documentation. As a standard part of a risk assessment all risks against the company must be considered. This includes threat actors who have the potential to cause harm to the company.

3.10.1 IS1

Within the Government world the standardised risk management form is known as the IS1. This is HMG Information Assurance Standard No 1 [1]. It is specific to government computer systems. The standard was developed by CESG and published and is available for anyone to adopt. The result of this has been wide spread takeup throughout the industry.

The standard itself defines a number of threat actors:

- Disaffected or dishonest employees

- Foreign Intelligence Services
- Amateur or professional hacker
- Virus and other malware writers
- Terrorist
- Investigative journalist
- Commercial Competitors
- Political Pressure groups/Activists
- Organised Criminal Groups

These specific terms are not defined within the documentation as they are only indicative of the threat actors. The standard goes further than simply listing the threat actors, stating that each of these can be classified under a number of ”families” of threat. The families break down as follows:

- System and Service Users
  - Privilidged User (PU)
  - Normal User (NU)
  - Service Consumer (SC)
  - Shared Service Suscriber (SSS)
- Direct Connections
  - Information Exchange Partner (IEP)
  - Service Provider (SP)
- Indirect Connections
  - Indirect Connection (IC)
- Supply Chain
  - Supplier (SUP)
  - Handler (HAN)
- Physical Present
  - Privilidged User (PU)

- Normal User (NU)
- Bystander(BY)
- Person Within Range (PWR)
- Physical Intruder (PI)

A Privileged User (PU) is someone who has the capability to legitimately install and modify applications, services, equipment and security defences. These would be commonly known as Administrators.

A Normal User (NU) is a legitimate user of the system who has no special privilidges.

A Service Consumer (SC) is someone who uses the services which you offer. This may or may not be a service for which the consumer must be registered. An example would be a website service.

A Shared Service Subscriber (SSS) would only apply when you are using a service that is critical to your operations, but that service is also used legitimately by other parties who may affect your supply of that service. An example would be the power grid. The Shared Service Subscriber would be other customers of the national grid whose actions could affect the service supply to you.

An Information Exchange Partner (IEP) is a partner with whom you share data as part of your business function. This data sharing may be through a physical connection or through media exchange. You are reliant on the integrity of the data being shared.

A Service Provider (SP) is someone who provides a service to you. This would include power, communications, databases, internet access, web-hosting etc.

An Indirectly Connected (IC) threat is someone who is attacking the network from an external location. This covers any connection from the internet.

A Supplier (SUP) is anybody within the supply chain who provides, maintains or otherwise has access to software or equipment.

A Handler (HAN) is anybody who has legitimate reason to have physical access to the equipment, however does not have any user access. This can include people who test equipment, repair equipment, dispose of obsolete equipment or generally transport equipment.

A Bystander (BY) is someone who is authorised to be in the same location as the equipment but has no requirement to handle the equipment in any way.

A Person Within Range (PWR) is anybody within range of the electronic emanations of equipment, who may be able to receive, interfere with or disrupt the correct emanations.

A Physical Intruder (PI) is someone who has no legitimate reason to be the area, who now has access to equipment.

It is clear that the IS1 provides a comprehensive list of threat actors in a well thought out structure. This level of depth may be considered to deep for some considering cyber security, as it includes a number of physical elements which are regularly left out of cyber security planning.

3.10.2 JSP440

Joint Service Publication 440 (JSP440) is the restricted security manual for the Ministry Of Defence [2]. It is a comprehensive document of security procedures which includes within it a number of threat actors. It should be noted that this document goes far beyond cyber security, including a number of elements that are not relevant to the question in hand. As a result not all of the threat actors suggest are relevant. The example list of potential attackers within JSP440 includes:

- System Users without the necessary clearance level for the data
- System Users without the necessary Special Access Approval
- System Users with no need to know the data
- Maintenance Staff
- Cleaners
- Journalists
- Investigators
- Third Party Contractors
- Foreign Intelligence Services or their agents
- Terrorists
- Extremists
- Competitors

These threat actors are not fully defined within the document, however they are fairly self explanatory. The document also focuses heavily on the number of potential attackers who may attempt to breach your systems. The theory is the greater the number of adversaries, the greater the potential that one of them will breach your security. It should be noted that this document is due to be superseded.

3.10.3 ICS-CERT

ICS-CERT is the Industrial Control Systems - Computer Emergency Response Team. This is an American public funded body whose mission it is to secure the countries Critical National Infrastructure (CNI). They provide a number of resources for the development of a secure cyber strategy which include with it a list of cyber threat actors. This list is as follows:

- Bot-Network Operators
- Criminal Groups
- Foreign Intelligence Services
- Hackers
- Insiders
- Phishers
- Spammers
- Spyware/Malware Authors
- Terrorist

Bot-Net Operators are defined as hackers who take over multiple machines in order to coordinate attacks.
Criminal Groups are defined as groups of attackers who seek to breach systems for monetary gain.
Foreign Intelligence Services use cyber tools to engage in information gathering and espionage activities.
Hackers are defined as anyone who breaks into networks for the thrill of the challenge or bragging rights within the hacker community.
Insiders are anyone within your own organisation who target your systems to gain unauthorised access to restricted information or cause damage to your systems. This is listed as the greatest source of computer crime.
Phishers are defined as anybody who executes a phishing attack in an attempt to steal identities or information for monetary gain.
Spammers distribute unsoliceted e-mail.
Spyware/Malware authors carry out attacks against an organisation with the intent infecting the host machine.
Terrorist seek to destroy, incapacitate or exploit critical infrastructure in order to threaten national security,

cause mass casualties and damage public morale.

The list originated from the Government Accountability Office (GOA), Department of Homeland Security (DHS) and was originally floated within the ”Role In Critical Infrastructue Protection (CIP) Cybersecurity”.[113]. This list has been widely adopted within America and as a result has greatly influenced considerations here.

3.11 Use Cases

Having considered the literature in this area we will now assess to what extent the literature is actually applied within industry. To perform this assessment a number of individuals within cyber security related jobs whom may have a requirement for the use of archetype threat actors have been interviewed. A number of job roles have been targeted across a number of sectors. In each case their function within the organisation and experience has been listed. A brief explanation as to what their job function entails with regards to how they use archetype threat actors. Finally a list of the actors which they use within their job function as well as their interpretation of that archetype. All interviewee's agreed to contribute on the understanding that they and the companies for which they work for remain anonymous.

3.11.1 Banking Cyber Threat Intelligence Analyst

The Cyber Threat Intelligence Analyst interviewed has been in post with the bank for approximately one year, however prior to this position he worked in threat analysis of physical systems and locations, with around 10 years experience in that role. The role involves a large amount of horizon scanning and open source intelligence gathering to provide in depth intelligence on current threats against the organisation. The level of depth on these reportings goes far beyond the scope of an archetype actor. Initial reports however are graded under on of the following headings;

- Script Kiddie
- Criminal Hacker
- Organised Criminal Gang
- Hacktivist
- Security Researcher
- Insider Threat

A script kiddie was defined as a low skilled hacker who had a limited number of tool that they don't fully understand. The script kiddie would attempt to use a tool and quickly loose interest if the attack didn't work. If they did manage a breach then they would boast about it in certain forums but were unlikely to do much damage. The criminal hacker was more advanced in their understanding of the tools and more persistent in there use. Should the criminal hacker gain entry their aim was to extract money. Organised criminal gangs were distinguished by the very fact that they are organised. Their attempts would be carefully co-ordinated such that multiple attacks would happen at the same time in an orchestrated manor. Should a weakness be found this would be exploited by the entire group very quickly, showing good communication skills. A hacktivist is an individual or group who through disruption attempt to raise the profile of their own cause. It maybe that they are actively targeting the bank or simply they have found an exploit that will work and we are a high value target that will get a lot of publicity. Unlike the previous hackers, hacktivist are extremely noisy in their attacks. The security researcher was defined as an individual who performs unique attacks for personal achievement. The attacks tend to be extremely innovative. The security researchers themselves tend to fall into two broad categories; Those who do it to gain notoriety and therefore publish without disclosure, and those who are more professional, conform to ethical disclosure and are not after the fame element. The security researcher category primarily arose from the horizon scanning element of the interviewee's work. Finally insider threat was defined as any employee of the bank who was deemed to be releasing too much information. Once again this refers to open source intelligence scanning effort.

3.11.2 Banking Chief Information Risk Officer

This interviewee is the Chief Information Risk Officer (CIRO) for a national bank with 22 years of experience in the industry. It should be stated that this is a different bank from the previous interviewee. He took an entirely different approach when considering the threat against his company. Due to the massive complexities in the current supply chain for his company he argued that the system was too complex to properly map and therefore manage. He argued that the underlying problem is that he is no longer in control of his company data, as there is too much reliance on external contractors and service providers over which he has little control. Rather than considering specific archetypes actor working against his company and all of his suppliers he classed the types of risk under specific headings. The headings are ranked in increasing impact and risk. The heading are;

- Disruption
- Fraud/Theft
- Destruction

- Brand

Against each of these headings he mapped the internal and external risks. Some of these risks are what I would consider Archetypes, where as some of the risks are specific attack types. This is indicative of the final result;

	Disruption	Fraud/Theft	Destruction	Brand
External	DDOS	Organised crime	Cyber Warfare	Reputational damage
Internal	Insider Threat	Data Theft	Disgruntled Employee	Data Leak

As the majority of these were risks rather than threat actors definitions were not sought.

3.11.3 Defence Contractor Emergency Response Team

The Defence Contractor interviewed was a member of his organisations Computer Emergency Response Team (CERT). His role is to respond to current and ongoing attacks to perform immediate mitigation. There is then a secondary phase where evidence is collected so that the event can be analysed at a later date and changes to standard operating procedures can be applied to stop the same thing happening again. Due to the sensitive nature of the businesses for whom he works, evidence is never collected to a legal standard, and the attackers are never prosecuted. When asked about the threat actors that he considers as part of an incident he responded that as a first response is to assess the skill level of the attacker. Very broadly these are split into four categories;

- Script Kiddie
- Criminal Hacker
- Organised Crime
- State or State Sponsored

A script kiddie was defined as an attacker who had access to a number of off the shelf tools and had some capacity to use them. They may have quite honed skills on a selection of tools, but have a limited selection at their disposal. They are very uncomfortable outside of their area of knowledge and regularly make silly mistakes.
A criminal hacker was defined as an individual who a refined skill set to perform an attack however they work as an individual. The skill level would be on par with that of a professional penetration tester. They have a capability of using a wide range of tools and selecting the best tool for a specific job. They are also capable of creating simple scripts and modify tools from their original specification to meet their current needs. Organised Criminals were defined as groups of skilled individuals working together towards

a common goal. Because there are a number of individuals involved, the skill level of any particular individual within a specific area can be extremely high, as opposed to a criminal hacker who has to be ”jack of all trades”. Organised criminals tend to do a great deal of intelligence gathering before an attack is attempted, and can change attack styles very quickly. Ultimately the organised criminal will want some kind of pay off so will limit the resources of an attack to what they expect the pay off to be. Finally the State or State Sponsored attackers were defined to be groups of highly skilled and highly motivated individuals with almost unlimited resources to carry out an attack. The attacks are performed over extended periods of time and can be highly sophisticated in their approach. These attackers are often referred to as Advanced Persistent Threats (APT) although the interviewee did not like this term. The goal of this type of attacker is long term compromise of the network to enable any future operations, and the continuous gathering of intelligence.

It is clear from the definitions provided that the motivation behind an attack is not considered important to the Computer Emergency Response Team. With this broad brush approach to skill level, a very quick assessment can be performed and the output will dictate the level of response from the CERT. The ultimate aim of the CERT to provide the correct response for the threat. The secondary phase of evidence collection feeds back intelligence of what indicators to consider to make that initial assessment in the future.

3.11.4 Government Risk Assessor

The Government Risk Assessor who was interviewed is a contractor on a major government project. They are currently involved in the design and delivery of a major government network. Within this job role he is creating the documentation, performing the risk assessments and developing the Cyber Security Operations Centre (CSOC) capability. He is formally accredited as a CLAS consultant.

When considering threat actors he refers to the Government Accountability Offices (GOA), Department of Homeland Security (DHS) list. For the purposes of creating a risk assessment this is a perfectly adequate list. The list includes;

- Bot-Network Operators
- Criminal Groups
- Foreign Intelligence Services
- Hackers
- Insiders
- Phishers

- Spammers
- Spyware/Malware Authors
- Terrorist

This list has been covered in depth earlier in this report and will not be expanded upon here. The interviewee stated that this was the list that was used for the creation of the risk assessment literature. He also stated that this is the rough outline that the CSOC would be using although they would hold much more accurate intelligence records on individual threats. The CSOC is a continuously developing capability and whilst the current list will continue to be used, there will always be an attempt for improvement.

3.12 Proposed Archetype Actors

Having considered all of the documentation and how this documentation has been interpreted within a number of real world examples the archetype threat actors that will be used for the framework will now be presented. In each case the actor has a title, given an in depth description and then provided a table of highlights with the key characteristics of that archetype actor. Where possible standard threat actor names have been used and keep the same meaning to avoid confusion.

3.12.1 Script Kiddie

A Script Kiddie is an unskilled hacker who has access to a limited number of off the shelf hacking tools. They may show some skill in the use of those tools but lack an overall understanding of how the tools actually work. They can be prone to making silly mistakes. If an action doesn't work they quickly get bored and move on to try something else. Their actions may appear random to an observer, as they try new tools without understanding what they do.

Actor	Script Kiddie
Skill	Low
Perseverance	Low
Motivation	Low
Characteristics	Little understanding leads to silly mistakes. Uses off the shelf tools

Table 3.1. Script Kiddie

3.12.2 Hacker

A hacker has progressed beyond the skill level of a script kiddie and can now use a wide variety of tools with some competency. They could be as proficient as a professional penetration tester. Very few mistakes are made during an attack. The hacker is able to move beyond simple tools and develop their own scripts. A criminal hackers motivation is money, so as long as the attack will harvest monetary value beyond the perceived effort the attacker will continue. The non-criminal hacker will be motivated by the thrill of the chase and the bragging rights within their community should the attack succeed. Higher value targets will hold greater bragging rights and thus the perseverance will increase.

Actor	Hacker
Skill	Med
Perseverance	Med
Motivation	Money/Thrill of the chase and bragging rights
Characteristics	skilled in a number of tools. Able to develop their own scripts.

Table 3.2. Hacker

3.12.3 Organised Crime

Organised crime groups of skilled individuals working together towards a common goal. The aim is to monetise the attack, so whilst the attack is deemed to be cost effective the criminal gang will continue their attack. The gang will work as a team, meaning that there are a number of highly skilled professionals within the team with their own specialisms. There is a great deal of intelligence gathering performed before an attack is attempted. The teams are well organised and can change an attack vector extremely quickly, reacting to information each other finds.

Actor	Organised Crime
Skill	High
Perseverance	Med
Motivation	Money
Characteristics	Good teamwork. Fast adapting.

Table 3.3. Organised Crime

3.12.4 State or State Sponsored

A state or state sponsored hacker will have almost unlimited resources at there disposal. They can work in teams extremely effectively. Some of the attacks can be extremely sophisticated, even implementing never

before seen attack vectors. The aim of current attacks is to obtain a foothold in a network and exfiltrate information, ideally without being noticed. A huge amount of intelligence can be gathered against an organisation before any physical attack has even begun. The attacks are designed to last for the long term. At the moment state action is focused on intelligence gathering, but that is not to say that in the future there will not be an offensive attack.

Actor	State or State Sponsored
Skill	High
Perseverance	High
Motivation	Exfiltration of intelligence, Potentially offensive attack
Characteristics	Highly sophisticated attacks aimed at long term compromise and exfiltration of intelligence

Table 3.4. State of State sponsored

3.12.5 Hacktivist

A Hacktivist is a hacker who uses their skills to put forward their own political views. The attacker could be of any skills capability, but that vast majority are very low skilled. Often the focus of the attack is on disruption in the form of DDOS or defacement.

Actor	Hacktivist
Skill	Varies
Perseverance	Low-Med
Motivation	To further their own political views.
Characteristics	Attacks aimed to further their political views. Attack is usually in the form of a DDOS.

Table 3.5. Hacktivist

3.12.6 Security Researcher

A security researcher will use their skill to find an exploit new vulnerabilities. A researcher will often focus on a specific area and so be extremely skilled within their own field. Security researchers broadly fall in to two groups; Those who do it for the challenge and abide by ethical disclosure or those who do it for the notoriety and publish unpatched vulnerabilities to say they got there first.

Actor	Security Researcher
Skill	High
Perseverance	High
Motivation	Prove their own skill/ gain notoriety
Characteristics	Extremely highly skilled in a specific area of interest.

Table 3.6. Security Researcher

3.12.7 Insider Threat

Insider Threat is anybody working within the organisation who then causes some form of compromise to the systems. This area is too large for a single heading and thus it has been split into a number of threat actors under the branch of insider threat.

3.12.7.1 Normal User

A Normal User would not ordinarily perform an attack on the network however a disgruntled employee with normal user access could cause damage through abusing this access. Attacks could include data theft, intentional introduction of a virus. Any attack will normally occur just before the employee involved quits.

Actor	Unprivileged User
Skill	Low
Perseverance	Low
Motivation	Frustration against the company/ Bribery
Characteristics	A disgruntled employee may obtain as much data as possible before leaving.

Table 3.7. Unpriliged User

3.12.7.2 Privileged User

An administrator has the capability of having a much greater impact on a network if they become disgruntled. They can perform the same attacks as a normal user (although they may have greater access to data), but beyond this they may reduce the security provisioning through disabling or removing critical programs. As an administrator their skill capability is likely to be much greater than that of an ordinary user. An administrator is also likely to have greater capability to cover their tracks.

3.12.7.3 WIMP

A WIMP stands for Well Intentioned but Misguided Person. This has become an accepted term for any actor who had no intention of causing a compromise, but through their actions has caused an issue. These

Actor	Privileged user
Skill	High
Perseverance	low
Motivation	Frustration against the company/ Bribery
Characteristics	A disgruntled employee may obtain as much data as possible before leaving.

Table 3.8. Priviledged User

actions could be intentional or unintentional. For example a WIMP may connect a USB memory stick to a system that has been infected with a virus. They had no intention of causing any damage to the system but through their actions the system has become infected. Equally the action may be intentional but the WIMP feels that it must be done in order to get on with their job. For example a company may have a policy which greatly restricts the number of websites which they are able to visit. The WIMP may feel that they need access to a greater range of websites within their job function and so they use an external proxy service that encrypts the traffic to them in order to get around the filter. Because that traffic is encrypted the usual security scanning systems for internet traffic are unable to function.

Because the actor at work in this case is unintentional, it is highly unlikely that they will use advanced techniques. It is usually that something is restricting their way of working and they are attempting to find a way around this. If the bypass takes longer than the current method then there is no point in using the bypass, and so not a great deal of effort will be put into finding a method that works.

Actor	WIMP (Well Intentioned but Misguided Person
Skill	Low
Perseverance	Low
Motivation	Med
Characteristics	An employee attempting to do their job bypassing security protocol

Table 3.9. WIMP

3.12.7.4 Bystander

Someone with authorised physical access but no actual user access to the systems. A typical example would be a cleaner or security guard. There are a number of attacks that can be performed with physical access including the ”evil maid attack”, installation of a keylogger or physical theft of a data device. These employees are low paid and may be open to bribery.

Actor	Bystander
Skill	Low
Perseverance	Low
Motivation	low
Characteristics	Opportunistic attack

Table 3.10. Bystander

3.12.7.5 Contractor

A contractor is not a standard member of staff, but may require access to your systems to perform their job function. Depending on the circumstances it is highly likely that the vetting of contractor staff is as high as an ordinary member of staff. Also it is a well known attack vector to embed contractors within an organisation that they wish to attack. The contractor can be of any skill level.

Actor	Contractor
Skill	Varies
Perseverance	Low
Motivation	Med
Characteristics	Short term insider threat.

Table 3.11. Contractor

3.12.8 Supply Chain

A compromise within the supply chain can be almost impossible to detect. Any individual component may have been compromised, right back to the factory state. Supply chains are long and complicated and almost impossible to follow back to source in a complex system.

Actor	Supply Chain
Skill	High
Perseverance	High
Motivation	Low
Characteristics	Almost impossible to detect

Table 3.12. Supply Chain

Chapter 4

Library Development

4.1 Introduction

This chapter will document the design process in developing a library for use in conjunction with the framework model, as well as showing the implemented libraries overlayed on the framework. It was highlighted through the literature review and in the the creation of the libraries that there was very little research performed on the attributes at a physical level. This is probably due to the localised nature of physical attributes, and the global nature of cyber attribution. Never the less there will be situations where physical attributes can be detected and will add to the overall attribution picture. The fact that this weakness was highlighted by the framework already verifies that the intended goal of ”highlighting intelligence gaps” has worked successfully.

4.2 Design Requirements

The libraries should be developed by an expert in their respective field. An individual library will be a collection of attributes relating to a tool or technology. Each attribute must have a clear description to explain what the attribute relates to. The expert will pick the best data type for each attribute. The expert may set confidence values based on experience. In a normal use case it is likely that the library contents will be completed by an operator of the tool or technology, and then passed to an analyst to build the complete attribution picture. The analyst will therefore obtain multiple libraries of information, collate the data, developing links where possible, and then perform further analysis on the data. Part of this analysis will be to see if two libraries can be conjoined to create a greater picture. As a result it is essential that the same attribute in different libraries must share a common name to enable the link to be created by the analyst. A simple example could be an IP address that has been detected by two different tools. This would enable us to add the attributes of both tools which will now provide a much richer picture.

In order for the library system to work it must meet the requirements defined below

- A library will refer to an individual technology or tool
- A library should be complete of all possible attributes for a specific tool or technology, to highlight intelligence gaps
- The same attribute in different libraries must share a common name/number

4.3 Designing a Library

The focus on development of a library is to capture all of the possible attributes that it is possible to extract from a tool or technology. The attribute must be something that is adjustable or has variance. This variance may be the result of a conscious decision, such as a user changing a setting, or it may be an unconscious decision, such as a setting the user is unaware of being left in its default position. It is hoped that if all of the attributes are collected then a unique fingerprint will be discovered, that can be tracked. The following process should be followed when developing a library:

- Create a list of all possible attributes for a tool or library
- Allocate each attribute against a layer of the model
- Check to see if the attribute appears in another library
- Use OR create attribute ID
- Assign attribute UID
- Provide an attribute name
- Assign a confidence value
- Assign a attribute type
- Add the attribute description

The expert should create a list of all possible attributes that are detectable within the tool or technology that they are expert on. The expert is expected to assess each layer of the framework in turn and allocate the relevant attributes. As with any model there are limitations so the expert will have to select the best fit for the attribute within the model. Once the list is complete all existing libraries must be consulted to see if there is already that attribute contained within them. This will dictate the UID (unique Identification Number) that each attribute will be allocated. The UID value is a unique identification for a specific attribute. The UID

is made up of three elements. The first element refers to the Library from which it is belongs. The second element refers to the layer of the model in which the attribute sits. This enables easy merging from multiple libraries. Finally the third element is an individual attribute identifier. Should multiple libraries include the same attribute then the same they will share the same attribute ID value (but would have different library ID numbers). As an example 3.5.17 would relate to library 3, layer 5, attribute 17. Now that each attribute has an entry the next stage is to allocate the confidence value, attribute type, attribute description and where applicable, allocate Minimum, Maximum or specialist values (such as fixed values or netmasks) for every attribute.

This entire process can be simplified to the following flow diagram:

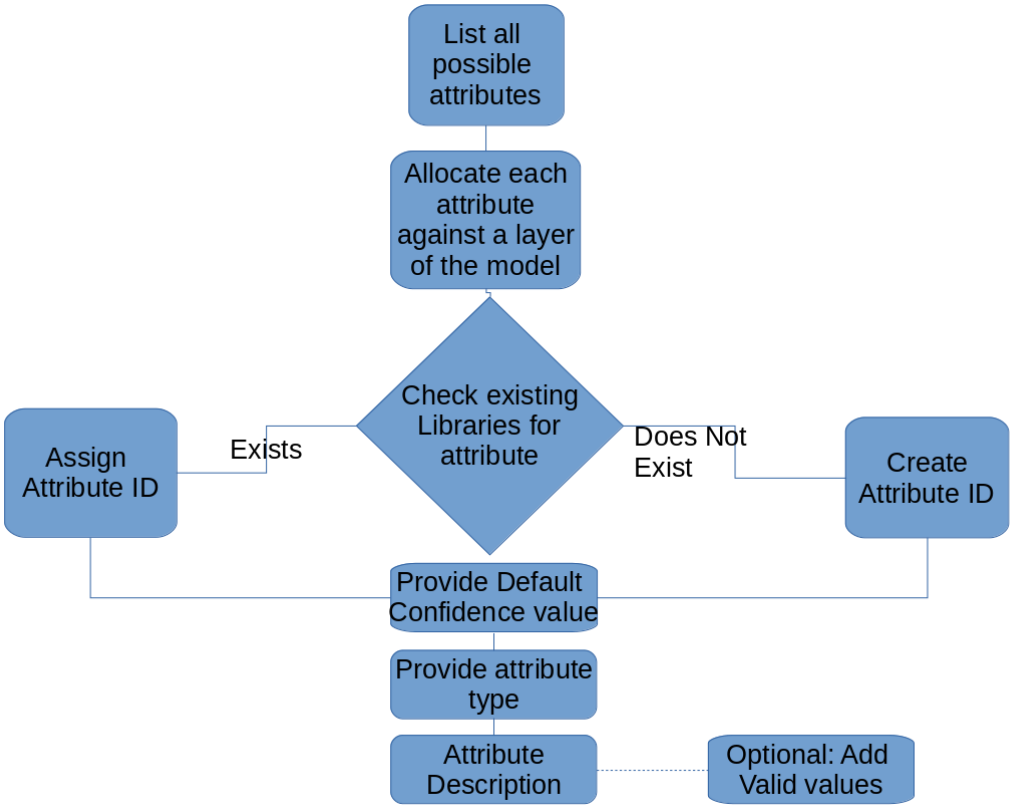

Fig. 4.1. Flow diagram for the creation of a library.

There are several advantages to this proposed method. From a development perspective this enables multiple experts to work together to build a single library, as each attribute is individually addressed. If a tool or technology is modified or updated it is easy to add to the library, creating a new UID for each new attribute. This will not not break the model. There is obviously clear scope for adding new libraries, making the model almost infinitely expandable and allowing the model to grow as new technologies and tools emerge. From an analyst perspective, with the numbering system employed by the attributes it is now easy to merge populated libraries to create an actor of multiple libraries. Comparison operations can

be performed on both attributes and libraries, and assist an analyst in deciding if the there is correlation between two libraries and/or actors.

4.4 Development of a library

As a demonstration of the process a library will now be created. It has been highlighted that there is currently a gap in the physical area of the framework. Several technologies exist within this layer which would by suitable for attribution analysis and could be incorporated into the framework. Examples would include Ethernet cabling (Both cat 5 and cat 6 would produce a library), fibre optic cabling and wireless technologies. For the purpose of demonstration wireless technologies will be analysed, focusing on the IEEE 802.11 standards. These are commonly referred to as home internet wireless. IEEE 802.11 is actually a set of standards dating back to July 1999 [3]. For the purpose of showing the development process a single standard will be used, however the process would be the same for all of the standards and any other library that is developed. In this example the 802.11g standard will be analysed. It must be stated that all of the measurements are based entirely on passive monitoring techniques and not compromising the wireless network security in order to remain within the confines of the law.

4.4.1 List all possible attributes

The first stage is to list all of the possible attributes that could be monitored passively on the 802.11g system. This is where the expert knowledge will really be of benefit, as there may be attributes that are a result of circumstances within the system rather than a feature of the technology. For each attribute in the example a rational behind the choice will be given. Ultimately the attributes have been place in the table ready for completion. This library has been developed from a full analysis of the standard [3] and an understanding of the signals processes involved.

4.4.1.1 RF elements

Fundamentally within the physical realm we are considering the Radio Frequencies being used for the transmission of data. These RF broadcasts can be measured in multiple ways. Each of these measurements will be an attributes

4.4.1.1.1 Amplitude The amplitude is a physical property of the wave being broadcast. It is fundamentally linked to the power of the broadcast. It can be manipulated to encode data. It is expected at a set distance from the source the amplitude will be a set size (the amplitude decreases with distance from

the source). As a result when you combine this information with a direction of travel it should be possible to estimate the location of the source. This is critical in discovering the physical location of an actor. The amplitude will be affected by a change in default power or a damaged antenna.

4.4.1.1.2 Frequency, Channels and Bandwidth The frequency will be the broadcast frequency that is being used for transmission. Due to it being an 802.11g device this will fall within the 2.4GHz band. By default the band is split into 14 channels, however country restrictions may limit the channels that are available. This restriction is usually applied within the firmware of the router. The channels are officially defined as having a bandwidth of 22MHz, however in in real world deployments this is normally set to 20MHz. This variance makes it a useful attribute as this is built into the router itself and not within the control of the user, so they are unable to manipulate it from a deception perspective. There is however enough variance across multiple manufacturers for detection. Because the frequency in use is actually an allocation of bandwidth the measurement has to be the frequency centre. Due to the cheap commercial hardware in use there is always some variance in the chips produced and this can result in frequency centre drift which is detectable as an attribute. The table below shows the channel, frequency centre and channel default bandwidth.

Channel	Frequency Center	Bandwidth (MHz)
1	2412	22
2	2417	24
3	2422	22
4	2427	22
5	2432	22
6	2437	22
7	2442	22
8	2447	22
9	2452	22
10	2457	17
11	2462	22
12	2467	22
13	2472	22
14	2477	22

4.4.1.1.3 Phase The phase is another fundamental property of an RF transmission and may be used to encode data.

4.4.1.1.4 Polarisation The polarisation of the signal is the ”direction” of the broadcast in 3 dimensional space. This is heavily influenced on the direction of the antenna in use. The orientation of the antenna, if not built into the router, is the decision of the user. Ordinarily a user would place the antenna straight up. If there is any angle to this it would be detectable. Additionally if there are multiple antenna’s in use by the router it is possible to detect this.

4.4.1.1.5 Modulation The modulation is how the RF signal is manipulated in order to encode data. In the 802.11g system there are three possible modulation methods available; CCK, DSSS and OFDM. CCK, Complementary Code Keying, is included in the standard to enable backwards comparability to the 802.11b standard. As a result it only offers speeds at 5.5 and 11 Mbps. DSSS, Direct Sequence Spread Spectrum and OFDM, Orthogonal Frequency Division Multiplexing methods can be combined in differing ways to get a range of different speeds.

Modulation Techniques	Use	Data Rates (Mbps)
DSSS and CCK	Mandatory	1, 2, 5.5, 11
OFDM	Mandatory	6,9,12,18,24,36,48,54
DSSS and PBCC (coding)	Optional	1,2,5.5,11,22,33
DSSS and OFDM	Optional	6,9,12,18,24,36,48,54

The critical point to note is the modulation techniques available result in different data rates. The available data rates are broadcast as a part of the beacon packet, a so it is possible to tell which modulation techniques are available on the router. More interestingly are situations where you can detect a specific modulation technique being used however not all of the data rates are offered. This can be an indication of firmware version. Another benefit of this attribute is that we have a set of discrete values which we can define in the library. These are :1, 2, 5.5, 6, 9, 11, 12, 18, 22, 24, 33, 36, 48, 54.

4.4.1.2 Frames

The physical radio waves are being used to encode data. In 802.11g the data is put into frames. There are a number of different varieties of frames with different roles however in a purely passive system it is only possible to capture the beacon frames. These are the frames that are broadcast by the wireless router to advertise its presence and the associated information required to enable a connection. Through the analysis of the frame contents it is possible to extract a number of attributes that may be used.

4.4.1.2.1 Beacon interval and Timing between beacons The Beacon Interval is the time between beacon broadcasts. This is a value given in ”time units” (TU). 1 time unit is 1024 microseconds. A default value of 100TU = 102.4 milliseconds. From an attribute perspective it is of interest if the value is something

other than the default of 100TU or if there is a discrepancy between the broadcast value and the recorded value in transmission. With the limited resources available on a wireless router there can be imperceivable delays in broadcast which are measurable. If the system is under stress then it is highly likely to be in use at the time which is critical information for a law enforcement agency.

4.4.1.2.2 Timestamp This is the time in milliseconds since the router was last powered off. This might provide useful supplementary information. For example if the router brand is known, a new firmware has been released for the device, however the time stamp dictates that the router has not been restarted in months then it is clear the the router is running the previous firmware version.

4.4.1.2.3 Capability Information The capability Information is actually a series of flags that can be set. Each flag will represent an attribute. The flags are:

- Immediate Block Ack not allowed
- Delayed Block Ack not allowed
- DSSS-OFDM is not allowed
- Radio measurement
- APSD is not supported
- G Mode short slot time 20 microseconds
- QoS is not supported
- Spectrum management is disabled
- Channel Agility Not Used
- PBCC not allowed
- Short Preamble not allowed
- Privacy disabled
- CF Poll not requested
- CF Not pollable
- Not an IBSS type network
- ESS Type network

4.4.1.2.4 SSID The SSID is the primary user defined field. This is the name of the wireless network that is broadcast and human readable. This can be any ASCII value upto 32 characters in length.

4.4.1.2.5 Supported Data Rates This was discussed at length previously. The advertised data rates can be compared to the modulation techniques in use. Also the range of offered data rates may be unique from the subset of:1, 2, 5.5, 6, 9, 11, 12, 18, 22, 24, 33, 36, 48, 54.

4.4.2 Optional elements of a beacon frame

All other elements of the beacon frame are optional. As a result there inclusion or lack of inclusion is an attribute in itself, as well as any values that have been set.

4.4.2.0.1 FH Parameter This is a legacy option only used by frequency hopping networks.

4.4.2.0.2 DS Parameter This is only present if the beacon is sent using a data rate of 1 or 2 MBps.

4.4.2.0.3 CF parameters These are co-ordinations Function Parameters which are used in conjunction with the Point Co-ordination Function. In reality these are never set in a real world deployment.

4.4.2.0.4 TIM TIM or Traffic Indication Map, is a method a router can use to inform a network that it has buffered broadcast messages

4.4.2.0.5 Country The country information broadcast is based on a country code. Additional information is broadcast including all of the channels that are available and the maximum broadcast power allowed on each channel.

4.4.2.0.6 FH Parameters and Pattern Table Once again this refers to Frequency Hopping and is a legacy capability that is not used on a modern router.

4.4.2.0.7 ERP Information ERP's are Extended Rate Physicals, and relate to methods of enhancing throughput. These are only available if the router supports it and there are no other interfering stations. This parameter dictates if ERP is enabled or not.

4.4.2.0.8 Extended Supported Rates This is an optional area which is only used if the router support more than 8 different data rates. The additional data rates that could not fit in the mandatory element of the beacon are placed here.

4.4.2.0.9 RSN The RSN element is the Robust Secure Network and provides information on the types of security that are in place on the network and any required settings. As a result there are several elements to the RSN settings which would count as attributes.

Element ID One byte in length, the element ID is the identifying the type of data to follow. In the case of RSN this is always 48. This means that it is not a suitable attribute.

Length One byte in length, this is the overall length of the RSN element. This will vary depending upon the PMK and Cipher Suites in use.

Version The version refers to the RSN version and is almost certainly 1, however it has the potential of being any numerical value stored by a byte.

Group Cipher Suite The Group Cipher Suite value fundamentally dictates the types of encryption that are available on an access point during broadcast and multicast transmissions. By default it will always offer the strongest encryption method first. The following are valid cipher values in order of strength, with the strongest method first:

- 00-0F-AC-04 (CCMP)
- 00-0F-AC-02 (TKIP)
- 00-0F-AC-03 (WRAP)
- 00-0F-AC-05 (WEP-104)
- 00-0F-AC-01 (WEP-40

Pairwise Cipher Suite Count The count dictates the number of Cipher's that are being offered by the access point for unicast transmissions.

Pairwise Cipher Suite This is cipher suite which is used for unicast broadcast. The same ciphers are available and once again the default is the strongest method.

Authentication Suite Count This count dictates the number of authentication methods that are offered by the access point. Currently this is a maximum of three.

Authentication Suite This is a list of the authentication methods that are available within the access platform. The number of entries here are dictated by the Authentication Suite Count. Possible values are:

- 00-0F-AC-01 (802.1X)
- 00-0F-AC-02 (PSK)
- 00-0F-AC-03 (FT Over 802.1X)

RSN Capabilities This is another flag area of 16 bits, providing an attribute for every flag.

- Reserved (0-1
- Extended Key ID for individually addressed frames (2)
- PBAC not supported (3)
- SPP A-MSDU Required not allowed (4)
- SPP A-MSDU Capable not supported (5)
- PeerKer Handshake Not Supported (6)
- Reserved (7)
- Managment Frame Protection Capable (8)
- Manament Frame Protection Required (9)
- GTKSA Replay Counter (10-11)
- PTSKA Replay Counter (12-13)
- Does not support ”No Pairwise” (14)
- Does not support Pre Authentication (15)

PMK Count The PMK count dictates the number of stored Pairwise Master Keys. This is a caching option to increase the speed of authentications as the EAP exchange does not need to take place.

PMK List This is a list of identifiers for Pairwise Master Keys that have been cached. This negates the need for an EAP exchange to swap PMK’s. The number of identifiers in the list is stated by the PMK count.

4.4.2.1 802.11g Attributes

This concludes the attributes that can be detected through passive means based on the 802.11g standard. For verification purposes this list could be checked by additional experts to highlight any errors. It is recommended however that this is performed after the next step.

4.4.3 Apply each attribute to a layer on the model

Now that all attributes have been listed, the next step in the process is to allocate them against the model, highlighting the layer in which they fit. For this purpose all attributes will be placed in the table and allocated a layer. The layer number has been placed in the UID, as it will form a part of the UID identification. Additionally the data types and valid attributes have

Table 4.1: IEEE 802.11g Attributes (9)

IEEE 802.11g Attributes (8)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
1	Amplitude	1		
1	Frequency	1		
1	Phase	1		
1	Channel	1		
1	Bandwidth	1		
1	Polarisation	1		
1	Modulation	1		
1	Timing between Beacons	1		
2	Broadcast Beacon Interval TTU	1		
2	Timestamp	1		
2	CI Immediate Block Ack not allowed	1		
2	CI Delayed Block Ack not allowed	1		
2	CI DSSS-OFDM is not allowed	1		
2	CI Radio measurement	1		
2	CI APSD is not supported	1		
2	CI G Mode short slot time 20 microseconds	1		

Continued on next page

Table 4.1 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
2	CI QoS is not supported	1		
2	CI Spectrum management is disabled	1		
2	CI Channel Agility Not Used	1		
2	CI PBCC not allowed	1		
2	CI Short Preamble not allowed	1		
2	CI Privacy disabled	1		
2	CI CF Poll not requested	1		
2	CI CF Not pollable	1		
2	CI Not an IBSS type network	1		
2	CI ESS Type network	1		
2	SSID	1		
2	Offered Data Rates	1		
2	Optional FH Parameter	1		
2	Optional DS Parameter	1		
2	Optional CF Parameter	1		
2	Optional Traffic Indication Map	1		
2	Optional Country	1		
2	Optional FH Parameters	1		

Continued on next page

Table 4.1 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
2	Optional FH Pattern Table	1		
2	Optional ERP Information	1		
2	Optional Extended Supported Rates	1		
2	Optional RSN Element ID	1		
2	Optional RSN Length	1		
2	Optional RSN Version	1		
2	Optional RSN Group Cipher Suite	1		
2	Optional RSN Pairwise Cipher Suite Count	1		
2	Optional RSN Pairwise Cipher Suite	1		
2	Optional RSN Authentication Suite Count	1		
2	Optional RSN Authentication Suite	1		
2	Optional RSN CAP Reserved (0-1)	1		
2	Optional RSN CAP Extended Key ID for individually addressed frames (2)	1		
2	Optional RSN CAP PBAC not supported (3)	1		
2	Optional RSN CAP SPP A-MSDU Required not allowed (4)	1		
2	Optional RSN CAP SPP A-MSDU Capable not supported (5)	1		
2	Optional RSN CAP PeerKer Handshake Not Supported (6)	1		
2	Optional RSN CAP Reserved (7)	1		

Continued on next page

Table 4.1 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
2	Optional RSN CAP Management Frame Protection Capable (8)	1		
2	Optional RSN CAP Management Frame Protection Required (9)	1		
2	Optional RSN CAP GTKSA Replay Counter (10-11)	1		
2	Optional RSN CAP PTSKA Replay Counter (12-13)	1		
2	Optional RSN CAP Does not support No Pairwise (14)	1		
2	Optional RSN CAP Does not support Pre Authentication (15)	1		
2	Optional PMK Count	1		
2	Optional PMK List	1		

Due to the legal restrictions of passive analysis it is not possible to proceed beyond the Data layer. As a result no further layers can be populated. It would be at this point in which other experts could be consulted to assess if all attributes have been captured. Additionally, with the allocation against the model it should highlight where the bulk of attributes sit and therefore any weaknesses with the current attribute set, which may be an indication of missed attributes.

4.4.4 Check attributes in existing libraries and apply UID

At this point the process is to check existing libraries for attributes which also exist within them. This would enable the correct UID to be allocated. On this occasion there are no attributes to be found in other libraries as the library is being created to fill an intelligence gap. This means that new UIDS can be created. The last number for each layer must be looked up from the last created library. In this case there are no layer 1 attributes, so numbering can start from 1, however there are layer 2 attributes and so numbering must start at 15. Finally to create the complete UID the library must be numbered, in this case the next available number is 8. With this information the table is populated.

4.4.5 Apply attribute type

Each attribute listed is now allocated an attribute type to ensure that the data being inputted is valid for the type of attribute. In addition to validation it will enable certain search and mathematical functions to be performed on the data. For example if there is a search for an IP address within a specified range, it is important that the data is interpreted in the form of an IP address. The defined types as explained in the previous chapter are:

- Text_String
- Integer
- Real_Numbers
- IPv4
- IPv6

In addition to attribute types there may be further restrictions to the data inputs. In this example the offered data rates would be an example of where further restrictions would apply. Whilst any integer would be valid data input there are a limited number of actual figures (1, 2, 5.5, 6, 9, 11, 12, 18, 22, 24, 33, 36, 48, 54). These are placed into the valid attribute section. This will enable more accurate functions to be developed for those attributes.

4.4.6 Completed 802.11g Library (8)

The result of the above process results in a completed library. It is now available for an operator to populate with data as they detect threat actors in order to pass on the information to an intelligence analyst. It is then the analysts job to link the library feeds to develop an overall threat actor picture.

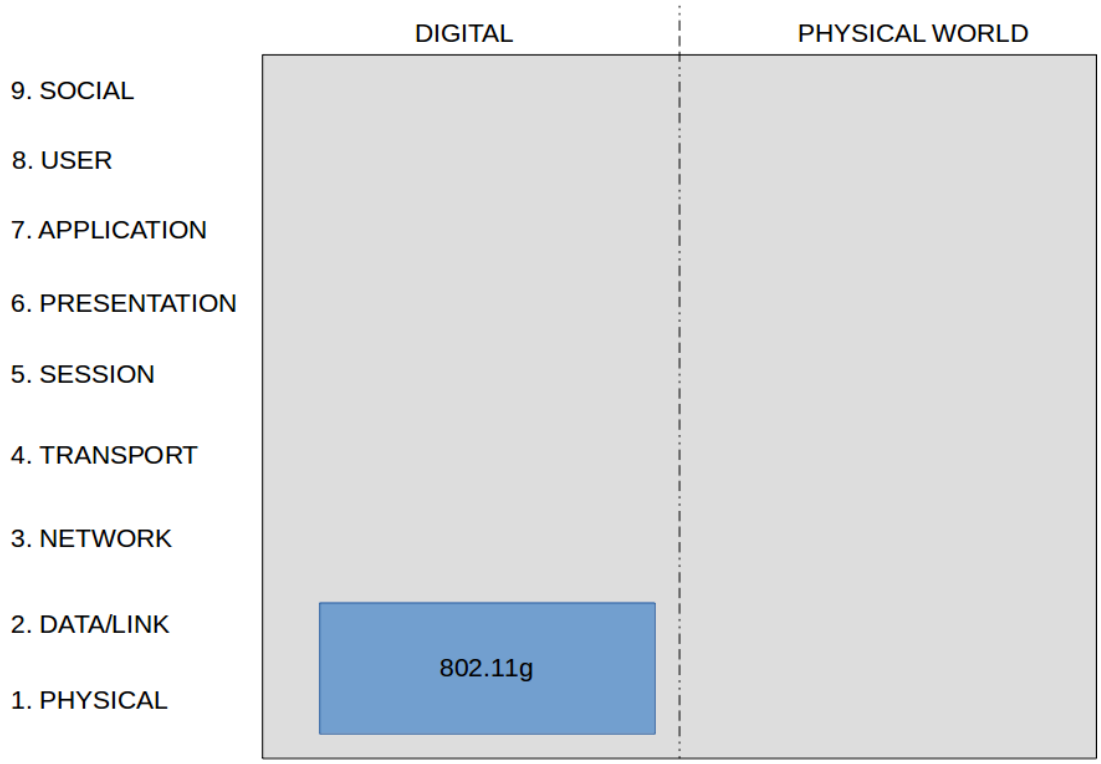

Fig. 4.2. 802.11g Library Coverage

Table 4.2: IEEE 802.11g Attributes (9)

IEEE 802.11g Attributes (8)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
9.1.1	Amplitude	1		
9.1.2	Frequency	1		
9.1.3	Phase	1		
9.1.4	Channel	1		1-14
9.1.5	Bandwidth	1		
9.1.6	Polarisation	1		
9.1.7	Modulation	1		CCK, DSSS, OFDM
9.1.8	Timing between Beacons	1		
9.2.15	Broadcast Beacon Interval TTU	1		
9.2.16	Timestamp	1		
9.2.17	CI Immediate Block Ack not allowed	1	True/false	0 OR 1
9.2.18	CI Delayed Block Ack not allowed	1	True/false	0 OR 1
9.2.19	CI DSSS-OFDM is not allowed	1	True/false	0 OR 1
9.2.20	CI Radio measurement	1	True/false	0 OR 1
9.2.21	CI APSD is not supported	1	True/false	0 OR 1
9.2.22	CI G Mode short slot time 20 microseconds	1	True/false	0 OR 1

Continued on next page

Table 4.2 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
9.2.23	CI QoS is not supported	1	True/false	0 OR 1
9.2.24	CI Spectrum management is disabled	1	True/false	0 OR 1
9.2.25	CI Channel Agility Not Used	1	True/false	0 OR 1
9.2.26	CI PBCC not allowed	1	True/false	0 OR 1
9.2.27	CI Short Preamble not allowed	1	True/false	0 OR 1
9.2.28	CI Privacy disabled	1	True/false	0 OR 1
9.2.29	CI CF Poll not requested	1	True/false	0 OR 1
9.2.30	CI CF Not pollable	1	True/false	0 OR 1
9.2.31	CI Not an IBSS type network	1	True/false	0 OR 1
9.2.32	CI ESS Type network	1	True/false	0 OR 1
9.2.33	SSID	1		
9.2.34	Offered Data Rates	1		1, 2, 5.5, 6, 9, 11, 12, 18, 22, 24, 33, 36, 48, 54
9.2.35	Optional FH Parameter	1		
9.2.36	Optional DS Parameter	1		
9.2.37	Optional CF Parameter	1		
9.2.38	Optional Traffic Indication Map	1		
9.2.39	Optional Country	1		

Continued on next page

Table 4.2 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
9.2.40	Optional FH Parameters	1		
9.2.41	Optional FH Pattern Table	1		
9.2.42	Optional ERP Information	1		
9.2.43	Optional Extended Supported Rates	1		1, 2, 5.5, 6, 9, 11, 12, 18, 22, 24, 33, 36, 48, 54
9.2.44	Optional RSN Element ID	1	Integer	48
9.2.45	Optional RSN Length	1		
9.2.46	Optional RSN Version	1		
9.2.47	Optional RSN Group Cipher Suite	1		00-0F-AC-01, 00-0F-AC-02, 00-0F-AC-03, 00-0F-AC-04, 00-0F-AC-05
2	Optional RSN Pairwise Cipher Suite Count	1		
9.2.48	Optional RSN Pairwise Cipher Suite	1		00-0F-AC-01, 00-0F-AC-02, 00-0F-AC-03, 00-0F-AC-04, 00-0F-AC-05
9.2.49	Optional RSN Authentication Suite Count	1		
9.2.50	Optional RSN Authentication Suite	1		00-0F-AC-01, 00-0F-AC-02, 00-0F-AC-03

Continued on next page

Table 4.2 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
9.2.51	Optional RSN CAP Reserved (0-1)	1		
9.2.52	Optional RSN CAP Extended Key ID for individually addressed frames (2)	1		
9.2.53	Optional RSN CAP PBAC not supported (3)	1		
9.2.54	Optional RSN CAP SPP A-MSDU Required not allowed (4)	1		
9.2.55	Optional RSN CAP SPP A-MSDU Capable not supported (5)	1		
9.2.56	Optional RSN CAP PeerKer Handshake Not Supported (6)	1		
9.2.57	Optional RSN CAP Reserved (7)	1		
9.2.58	Optional RSN CAP Management Frame Protection Capable (8)	1		
9.2.59	Optional RSN CAP Management Frame Protection Required (9)	1		
9.2.60	Optional RSN CAP GTKSA Replay Counter (10-11)	1		
9.2.61	Optional RSN CAP PTSKA Replay Counter (12-13)	1		
9.2.62	Optional RSN CAP Does not support No Pairwise (14)	1		
9.2.63	Optional RSN CAP Does not support Pre Authentication (15)	1		
9.2.64	Optional PMK Count	1		
9.2.65	Optional PMK List	1		

4.5 Further libraries implemented

Now that the library creation process has been demonstrated, the process has been applied to create develop several other libraries covering a range of technologies or tools. These have been created through analysis of the literature and extracting the attributes. In each case the literature used has been explained and where gaps in the library have been identified through use of the model, multiple sources have been used. The following libraries have been developed to prove the concepts of the model:

- IPv4
- Browser_Attributes
- Executables
- E-Mail
- Linguistic
- Physical_World

4.5.1 IPv4 (1)

The IP address is the primary method of recognition in any cyber incident. Although it is relatively easy to falsify the IP address through a system of proxies, ultimately that IP address is still an artefact of the attack and makes up a part of the fingerprint for the threat actor, even if it is not their actual IP address. Due to the majority of incident response and monitoring tools being based upon network traffic it is the IP address that is the most useful piece of data to be shared so that other monitoring tools can hone in on the potentially malicious traffic and the response teams can begin to block that traffic. Because it is so critical to a fast time investigation the IP address and related information has been designated as the first and primary library, as it is expected that this will be the linchpin linking the majority of the other libraries. Due to its importance the IP address and port numbers have been placed at the top of the library for ease. At this point only IPv4 has been modelled for testing as this represents the bulk of traffic but it is recognised that IPv6 will have to be implemented as future work.

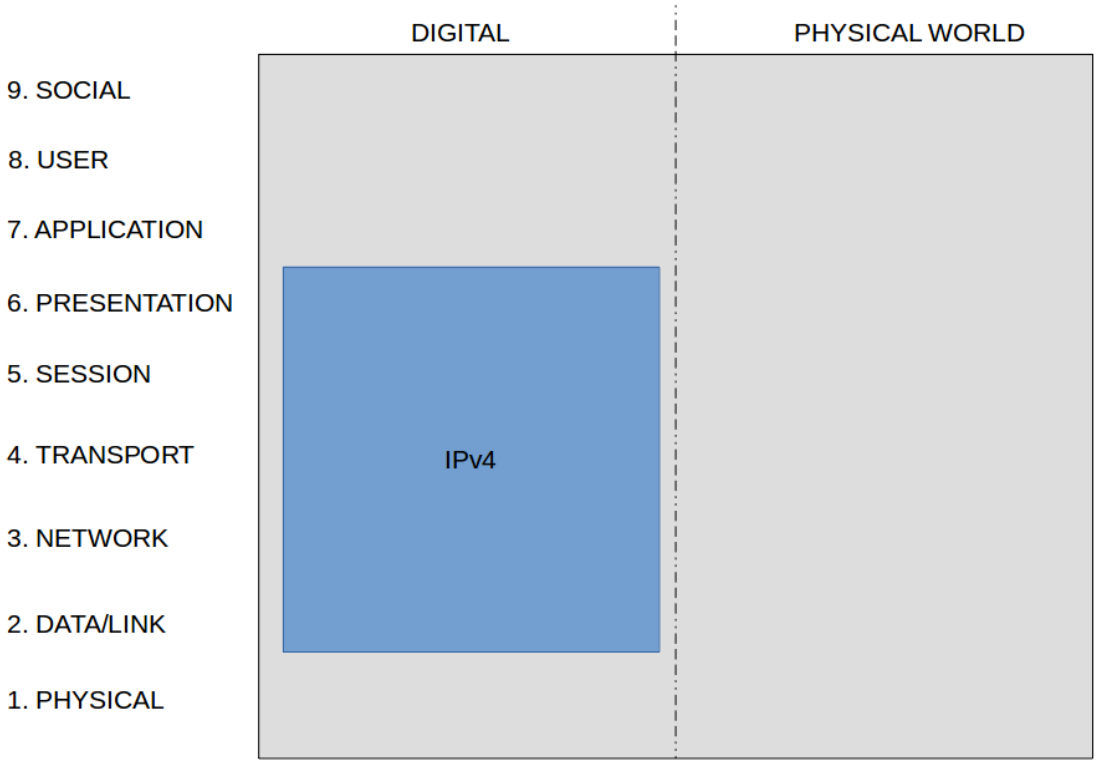

Fig. 4.3. IPv4 Library Coverage

Table 4.3: IPv4 (1)

IPv4 (1)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
1.3.1	Source IP address	1	IPv4	
1.3.2	Destination IP address	1	IPv4	
1.5.1	Source Port	1	Integer	0-65535
1.5.2	Destination Port	1	Integer	0-65535
1.2.1	Frame Number	1	Integer	0-254
1.2.2	Frame Length	1	Integer	
1.2.3	Destination MAC address	1	Text_String	
1.2.4	Source MAC address	1	Text_String	
1.3.3	IP Version	1	Integer	0-15 (expecting 4)
1.3.4	IP Header Length	1	Integer	0-15 (expecting 5)
1.3.5	Differentiated Services Codepoint	1	Integer	0-63 (expecting 0)
1.3.6	Explicit Congestion Notification	1	Integer	0-3 (expecting 0)
1.3.7	Total Length	1	Integer	
1.3.8	Fragmentation Flags - Reserved Bit	1	True/False	0 OR 1 (Expecting 0)
1.3.9	Fragmentation Flags - Don’t fragment	1	True/False	0 OR 1 (Expecting 1)
1.3.10	Fragmentation Flags - More fragments	1	True/False	0 OR 1 (Expecting 0)

Continued on next page

Table 4.3 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
1.3.11	Fragment offset	1	Integer	(Expecting 0)
1.3.12	TTL	1	Integer	1-254
1.3.13	Protocol	1	Integer	0-15
1.4.3	Sequence Number	1	Integer	
1.4.4	Acknowledgement number	1	Integer	
1.4.5	Header Length	1	Integer	
1.4.6	TCP flags - Reserved (1)	1	True/False	0 OR 1 (Expecting 0)
1.4.7	TCP flags - Reserved (2)	1	True/False	0 OR 1 (Expecting 0)
1.4.8	TCP flags - Reserved (3)	1	True/False	0 OR 1 (Expecting 0)
1.4.9	TCP flags - Nonce	1	True/False	0 OR 1
1.4.10	TCP flags - Congestion Window Reduced	1	True/False	0 OR 1
1.4.11	TCP flags - Echo	1	True/False	0 OR 1
1.4.12	TCP flags - Urgent	1	True/False	0 OR 1
1.4.13	TCP flags - Acknowledgement	1	True/False	0 OR 1
1.4.14	TCP flags - Push	1	True/False	0 OR 1
1.4.15	TCP flags - Reset	1	True/False	0 OR 1
1.4.16	TCP flags - Syn	1	True/False	0 OR 1
1.4.17	TCP flags - Fin	1	True/False	0 OR 1

Continued on next page

Table 4.3 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
1.4.18	Window Size Value	1	Integer	
1.4.19	Urgent Pointer	1	Integer	

4.5.2 Passive Operating system Fingerprinting (POF) (2)

The POF library is an example of a tool library. POF is the Passive Operating system Fingerprinting tool, developed by Michal Zalewski [205]. Once again the reliance is upon passive data rather than active probing to remain within the law. Through careful analysis of network traffic it is possible to guess at the remote operating system in use.

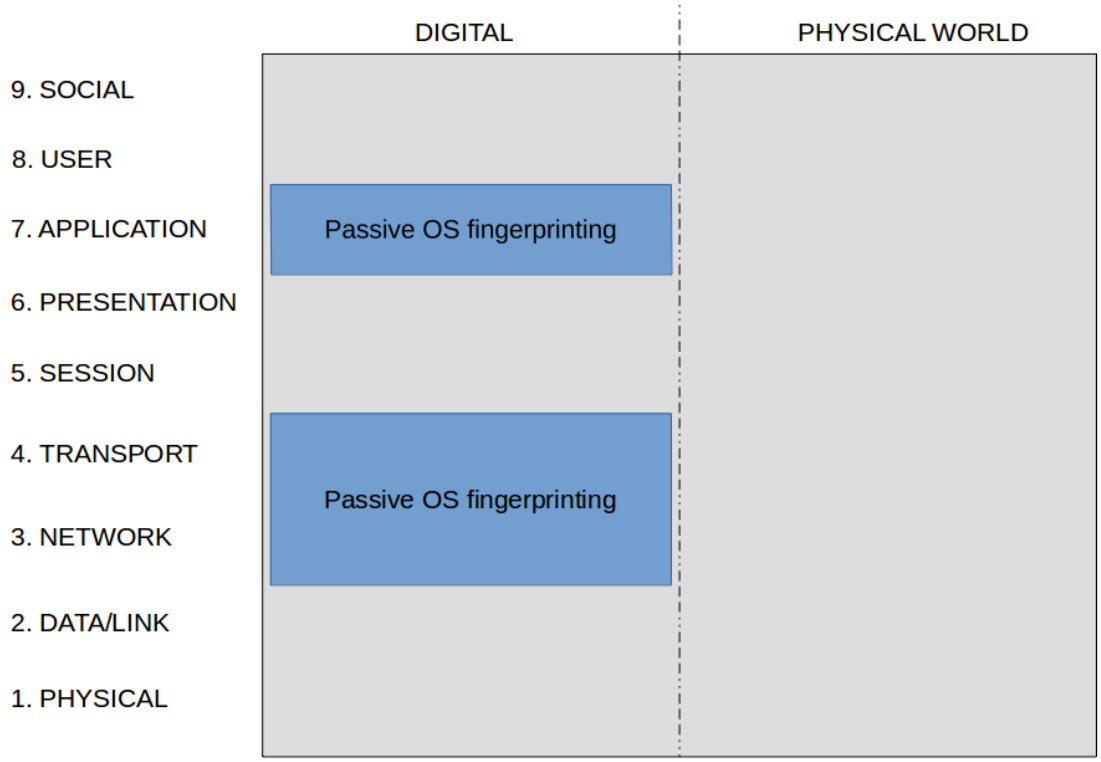

Fig. 4.4. Passive Operating system Fingerprinting Library Coverage

Table 4.4: Passive Operating System Fingerprinting (2)

Passive Operating System Fingerprinting (2)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
2.3.12	TTL	1	Integer	1-255
2.4.18	TCP Window Size	1	Integer	
2.3.9	Don't Fragment Flag	1	True/false	0 OR 1
2.3.14	TOS Minimize Delays	1	True/false	0 OR 1
2.3.15	TOS Maximise throughput	1	True/false	0 OR 1
2.3.16	TOS Maximise Reliability	1	True/false	0 OR 1
2.3.17	TOS Minimise monetary cost	1	True/false	0 OR 1
2.4.16	TCP SYN	1	True/false	0 OR 1
2.4.20	TCP SYN+ACK	1	True/false	0 OR 1
2.7.1	HTTP Request	1	True/false	0 OR 1
2.7.2	HTTP Response	1	True/false	0 OR 1

4.5.3 Browser (3)

The browser is the primary way in which a user is expected to interact with a web page. If you are the owner of the web page then a number of methods are available to you to attempt to track and fingerprint all of those who connect to you. It is common practice as a part of a cyber attack that during the reconnaissance phase of the attack the web page would be browsed. Whilst it may prove almost impossible to tie together the accessing of the website to an attack during the event, if the data has been captured it may be possible to fingerprint the browser after seizure in a criminal case. This will greatly improve the case against the defendant. The attributes used are based upon the EFF panoptoclick study [52]. This is the first library at the application layer.

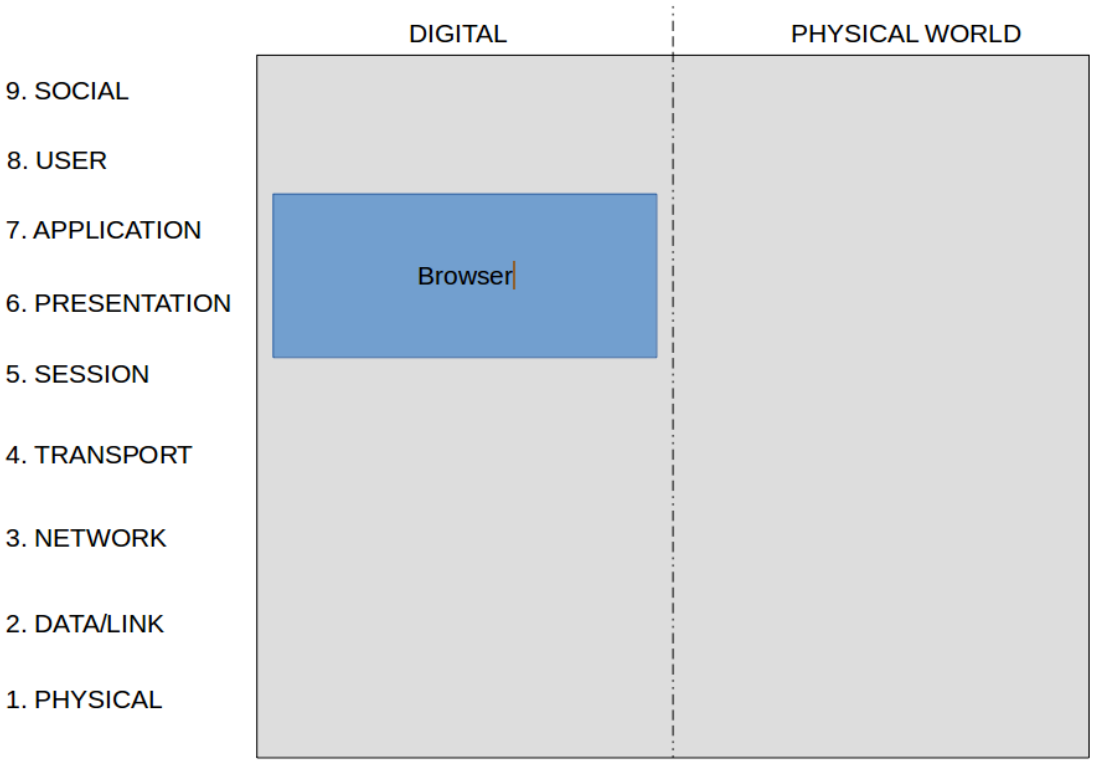

Fig. 4.5. Browser Library Coverage

Table 4.5: Browser (3)

Browser (3)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
3.7.3	User Agent	1	Text_String	
3.7.4	HTTP_ACCEPT Headers	1	Text_String	
3.7.5	Browser Plugin Details	1	Text_String	
3.7.6	Time Zone	1	Text_String	
3.6.1	Screen Size Horizontal	1	Integer	
3.6.2	Screen Size Vertical	1	Integer	
3.6.3	Screen Colour Depth	1	Integer	
3.6.4	Install System Fonts	1	Text_String	
3.7.7	Cookies	1	Text_String	
3.7.8	HTML5 GPS Location call	1	Text_String	
3.7.9	HTML5 SSID Location Call	1	Text_String	
3.6.5	WEBGL Graphics Request	1	Text_String	
3.7.10	Browser Cache	1	Text_String	

4.5.4 Executable Malware (4)

One of the primary methods for a cyber attack is the deployment of malware. This usually occurs in several stages, with an initial ”hook” calling back to a server to download further payloads or instructions. The payload would greatly depend on the motives of that attacker and links in to the threat actor assessment in the next chapter.

Whilst in a live and real time situation it is not possible to perform an in depth analysis on a potential piece of malware, it is possible to perform a superficial scan of the file which can be used to create a simplistic fingerprint. This can then be compared against other detected malware in an attempt to discover linked payloads. This is probably the most critical analysis in order to discover the likely archetype actor working against you. The payload will give an indication to intent.

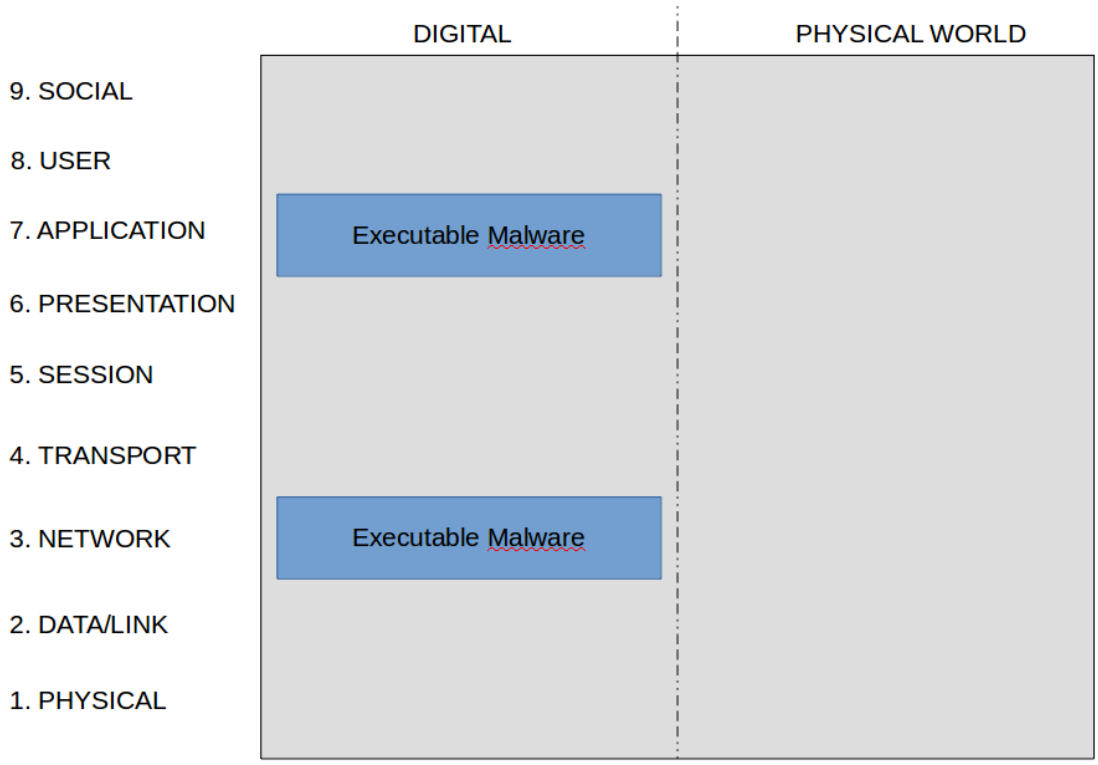

Fig. 4.6. Executable Malware Library Coverage

Table 4.6: Executable Malware (4)

Executable Malware (4)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
4.7.30	Debug Flag	1	True/false	0 OR 1
4.7.31	Bytes Reversed Lo Flag	1	True/false	0 OR 1
4.7.32	Bytes Reversed Hi Flag	1	True/false	0 OR 1
4.7.33	Relocs Stripped Flag	1	True/false	0 OR 1
4.3.1	IP address	1	IPv4	
4.3.18	Malware Domains	1	Text_String	
4.7.34	Malware Email	1	Text_String	
4.7.35	Bitcoin Address	1	Text_String	
4.7.36	Library Imports	1	Text_String	
4.7.37	Dropped Files	1	Text_String	
4.7.38	Packer	1	Text_String	
4.7.39	Entropy	1	Text_String	
4.7.40	Compilation Date	1	Text_String	
4.7.41	Malware Activity - Windows Registry	1	True/false	0 OR 1
4.7.42	Malware Activity - Takes Screenshot	1	True/false	0 OR 1
4.7.43	Malware Activity - Windows Files Operations	1	True/false	0 OR 1

Continued on next page

Table 4.6 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
4.7.44	Malware Activity - Affects Private Profile	1	True/false	0 OR 1
4.7.45	Malware Activity - Privilege Escalation	1	True/false	0 OR 1
4.7.46	Malware Activity - Affect System Token	1	True/false	0 OR 1
4.7.47	Malware Activity - Windows Mutex	1	True/false	0 OR 1
4.7.48	Malware Activity - Keylogger	1	True/false	0 OR 1
4.7.49	Malware Activity - Windows Hook	1	True/false	0 OR 1
4.7.50	Malware Activity - RAW TCP socket communications	1	True/false	0 OR 1
4.7.51	Malware Activity - Webcam Access	1	True/false	0 OR 1
4.7.52	Malware Activity - TCP Listen	1	True/false	0 OR 1
4.7.53	Malware Activity - UDP communications	1	True/false	0 OR 1
4.7.54	Malware Activity - DNS communications	1	True/false	0 OR 1
4.7.55	Malware Activity - SMTP RAW communications	1	True/false	0 OR 1
4.7.55	Malware Activity - Disable Registry Editor	1	True/false	0 OR 1

- Text_String
- Integer
- Real_Numbers
- IPv4
- IPv6
- True/false

4.5.5 E-Mail (5)

Whilst it is recognised that the majority of this information can be easily spoofed, should the attacker use the same spoofed information then, although the information is incorrect, the fingerprint still exists, and builds upon the attribution picture. Through analysis of the emails it may be possible to work out an attack pattern, and possible targets based upon who the emails have been sent to. As a result a fingerprint is critical to gaining this insight.

This library is based upon the work of De Vel et al. [46] and Gupta et al. [69].

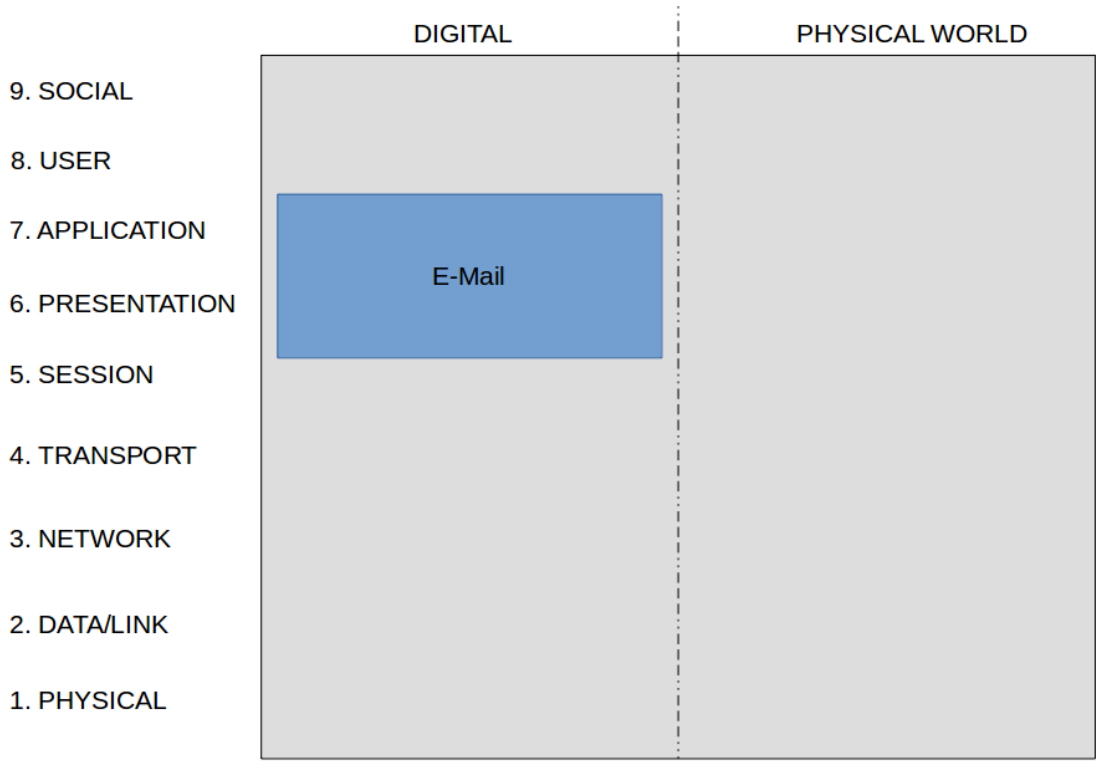

Fig. 4.7. E-Mail Library Coverage

Table 4.7: E-Mail (5)

E-Mail (5)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
5.7.11	Mime Version	1	Integer	
5.7.12	Time Zone	1	Integer	
5.7.13	Message-ID	1	Text_String	
5.7.14	Subject	1	Text_String	
5.7.15	To	1	Text_String	
5.7.16	From	1	Text_String	
5.7.17	Content Type	1	Text_String	
5.7.18	Thread Index	1	Text_String	
5.7.19	Accept Language	1	Text_String	
5.8.1	Content Language	1	Text_String	
5.7.20	Authentication Result	1	Text_String	
5.7.21	References	1	Text_String	
5.7.22	Attachments	1	Text_String	
5.6.6	Content Transfer Encoding	1	Text_String	

4.5.6 Social Media (15)

A generic social media library was developed during the experimentation phase whilst performing a live task. Due to the strict time restrictions involved in the task not all attributes have been carefully considered, and the attributes included at present are only the ones that were useful for the specified task. A full analysis is required for each social media platform however this simplest library is suitable for basic tasks. The following were used for Twitter, Instagram, Facebook and Youtube.

- Username
- Groups Joined
- Joining Date

The Username is critical as this is the primary identifying information and people will tend to use the same user name across multiple platforms. The username may well be a pseudonym so every effort must be made to tie that pseudonym to a real world identity. The groups joined are useful in identifying groups of threat actor who are working together. A further example of this may be a hacking forum where multiple users work together to achieve an attack. This is the first example inside layer 9 of the model. It should also be stated that these social grouping may have an existence in the real world which should be investigated. Joining date is useful to identify false identities, whilst still tying together the accounts. If all of the accounts across all social media platforms were created on the same day, then that is a very good indication of false information, whilst still attributing it to a single threat actor.

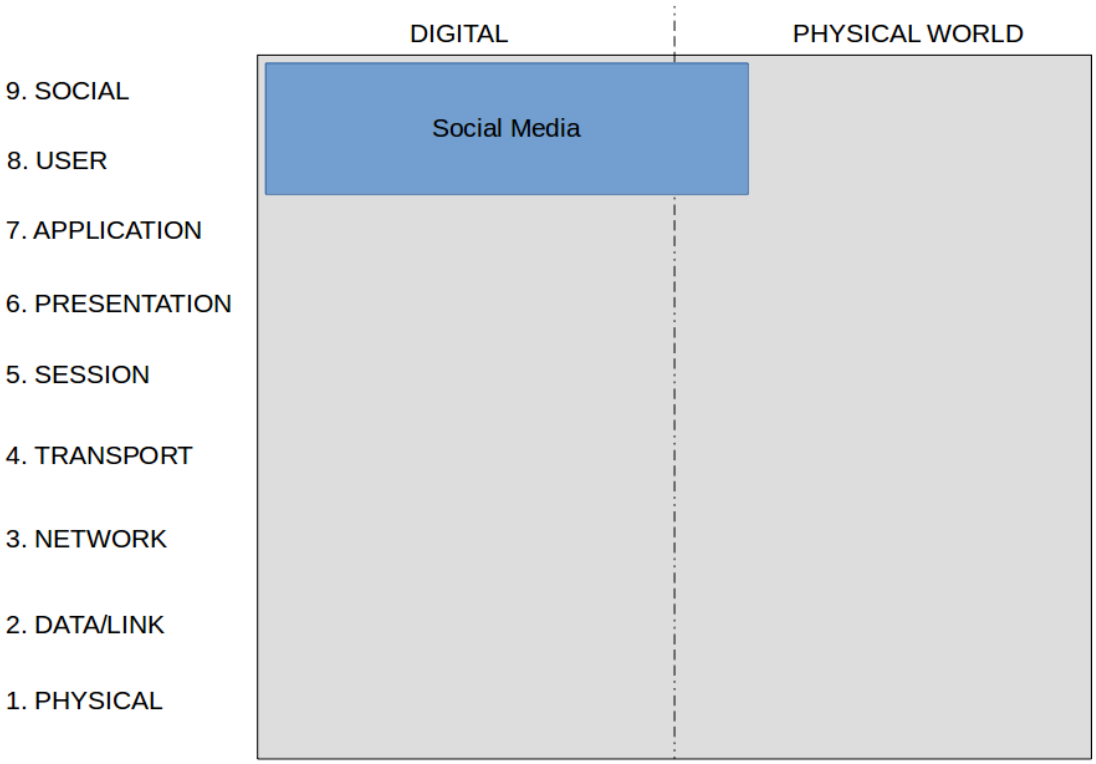

Fig. 4.8. Social Media Library Coverage

Table 4.8: Social Media (15)

Social Media (15)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
15.8.120	Username	1	String	
15.8.121	Joining Date	1	String	
15.9.1	Groups Joined	1	String	

4.5.7 Linguistic

The linguistic elements sit within layers 8 and 9 of the model. Primarily the analysis would be performed on an individual, and their own unique linguistic style will be fingerprinted. This is useful in evidence as well as attempting to tie together several disparate pieces of information. There is however a social element when a group working together will use certain key words, or use the same misspelling of a word [136]. This would enable the linking of multiple threat actors within a single campaign. Due to the scale of the lexical library it has been broken down three separate libraries, Lexical attributes, Syntactical attributes and structural attributes. In reality a single analyst would be able to provide data for all libraries.

These libraries have been created based upon the extensive research documented in chapter two.

4.5.7.1 Lexical Attributes (6)

The lexical attributes relate to the key features of a document from either a character or a word perspective. It is important when using this type of analysis that you only use media of the same type. For example you may compare forums postings, or you may compare tweets, however you should never compare the two types of media in direct comparison.

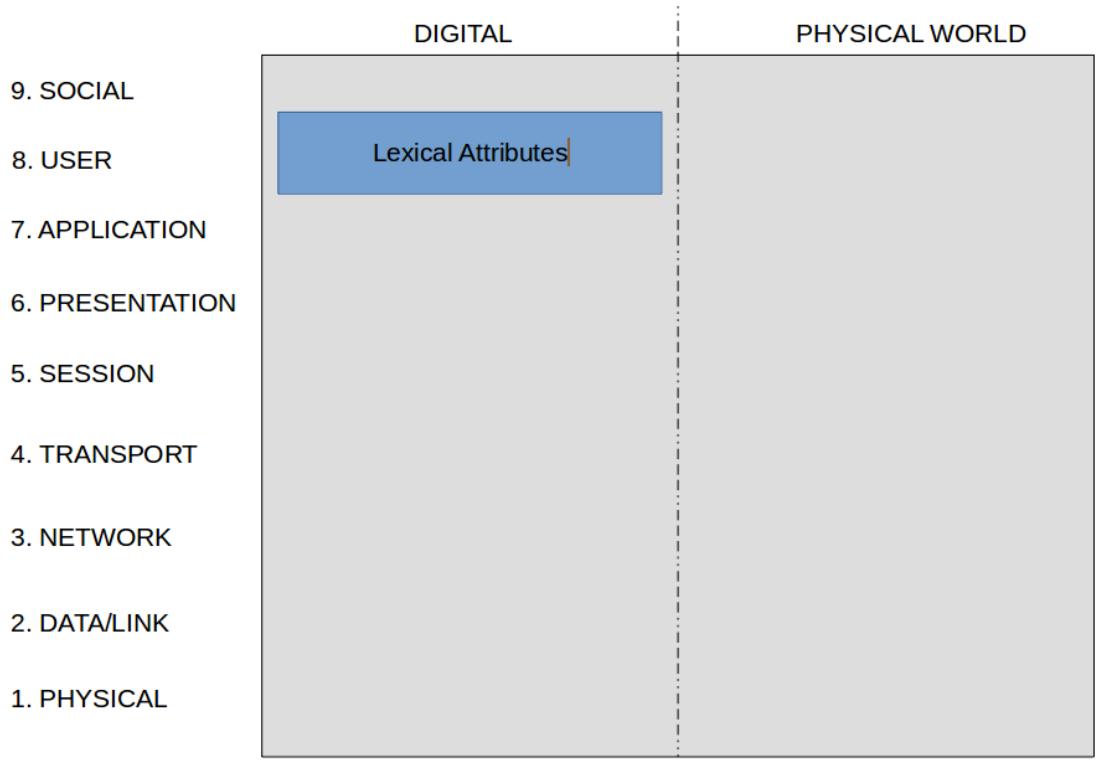

Fig. 4.9. Lexical Linguistic Library Coverage

Table 4.9: Lexical Attributes (6)

Lexical Attributes (6)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
Character Features				
6.8.2	Total number of Character (C)	1	Integer	
6.8.3	Total number of alphabetic Characters/C	1	Integer	
6.8.4	Total number of upper case characters/C	1	Integer	
6.8.5	Total number of digit characters/C	1	Integer	
6.8.6	Total number of white space characters/C	1	Integer	
6.8.7	Total number of tab spaces/C	1	Integer	
6.8.8	Frequency of A	1	Integer	
6.8.9	Frequency of B	1	Integer	
6.8.10	Frequency of C	1	Integer	
6.8.11	Frequency of D	1	Integer	
6.8.12	Frequency of E	1	Integer	
6.8.13	Frequency of F	1	Integer	
6.8.14	Frequency of G	1	Integer	
6.8.15	Frequency of H	1	Integer	
6.8.16	Frequency of I	1	Integer	

Continued on next page

Table 4.9 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
6.8.17	Frequency of J	1	Integer	
6.8.18	Frequency of K	1	Integer	
6.8.19	Frequency of L	1	Integer	
6.8.20	Frequency of M	1	Integer	
6.8.21	Frequency of N	1	Integer	
6.8.22	Frequency of O	1	Integer	
6.8.23	Frequency of P	1	Integer	
6.8.24	Frequency of Q	1	Integer	
6.8.25	Frequency of R	1	Integer	
6.8.26	Frequency of S	1	Integer	
6.8.27	Frequency of T	1	Integer	
6.8.28	Frequency of U	1	Integer	
6.8.29	Frequency of V	1	Integer	
6.8.30	Frequency of W	1	Integer	
6.8.31	Frequency of X	1	Integer	
6.8.32	Frequency of Y	1	Integer	
6.8.33	Frequency of Z	1	Integer	
6.8.34	Frequency of @	1	Integer	

Continued on next page

Table 4.9 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
6.8.35	Frequency of #	1	Integer	
6.8.36	Frequency of $	1	Integer	
6.8.37	Frequency of *	1	Integer	
6.8.38	Frequency of &	1	Integer	
6.8.39	Frequency of ”	1	Integer	
6.8.40	Frequency of	1	Integer	
6.8.41	Frequency of _	1	Integer	
6.8.42	Frequency of =	1	Integer	
6.8.43	Frequency of +	1	Integer	
6.8.44	Frequency of ¡	1	Integer	
6.8.45	Frequency of ¿	1	Integer	
6.8.46	Frequency of [	1	Integer	
6.8.47	Frequency of]	1	Integer	
6.8.48	Frequency of {	1	Integer	
6.8.49	Frequency of }	1	Integer	
6.8.50	Frequency of :	1	Integer	
Word Features				
6.8.60	Total number of words (M)	1	Integer	

Continued on next page

Table 4.9 – *Continued from previous page*

UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
6.8.61	Total number of short words (less than 4 chars)/M	1	Integer	
6.8.62	Total number of characters in words/C	1	Integer	
6.8.63	Average word length	1	Integer	
6.8.64	Average sentence length in terms of characters	1	Integer	
6.8.65	Average sentence length in terms of words	1	Integer	
6.8.66	Total different words/M	1	Integer	
6.8.67	Hapax legomena	1	Integer	
6.8.68	Hapax dislegomena	1	Integer	
6.8.69	Yule's K measure	1	Integer	
6.8.70	Simpson's D measure	1	Integer	
6.8.71	Sichel's S measure	1	Integer	
6.8.72	Brunet's W measure	1	Integer	
6.8.73	Honore's R measure	1	Integer	

4.5.7.2 Syntactical attributes (7)

Syntactical analysis is looking at the choices that an author has made at specific points in a text. Did the author choose to start a new sentence, or continue the sentence with the use of a comma. Has the author used a colon or a semicolon? These are all decisions ultimately taken by the author, and representing the thought flow of that author. Additionally to punctuation included in the model it would be possible to include the frequency of specific function words. Did the author use male or female function words? Did they choose to use in over on? The automated analysis of function words is extremely difficult and so has not been included in the model at this time.

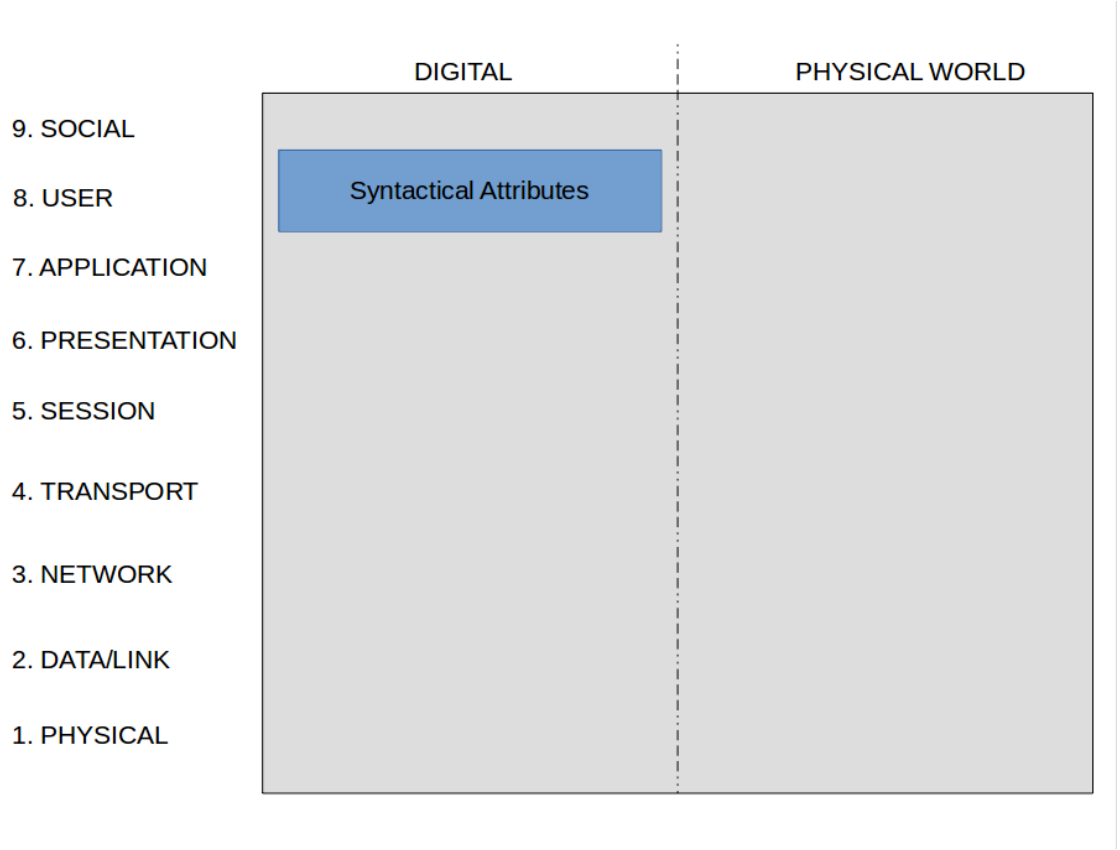

Fig. 4.10. Syntactical Linguistic Library Coverage

Table 4.10: Syntactical attributes (7)

Syntactical attributes (7)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
Character Features				
6.8.80	Frequency of punctuation .	1	Integer	
6.8.81	Frequency of punctuation ,	1	Integer	
6.8.82	Frequency of punctuation ?	1	Integer	
6.8.83	Frequency of punctuation !	1	Integer	
6.8.84	Frequency of punctuation :	1	Integer	
6.8.85	Frequency of punctuation ;	1	Integer	
6.8.86	Frequency of punctuation ’	1	Integer	
6.8.87	Frequency of punctuation ”	1	Integer	

4.5.7.3 Structural attributes (8)

Structural attributes relate to an authors choice in the layout and building blocks of the text. This can be very generic or it can be specific to a type of media. For this example there has been the inclusion of E-Mail however this could be extended to any media type. For example, in the analysis of tweets you could analyse the use of URL shortening service used.

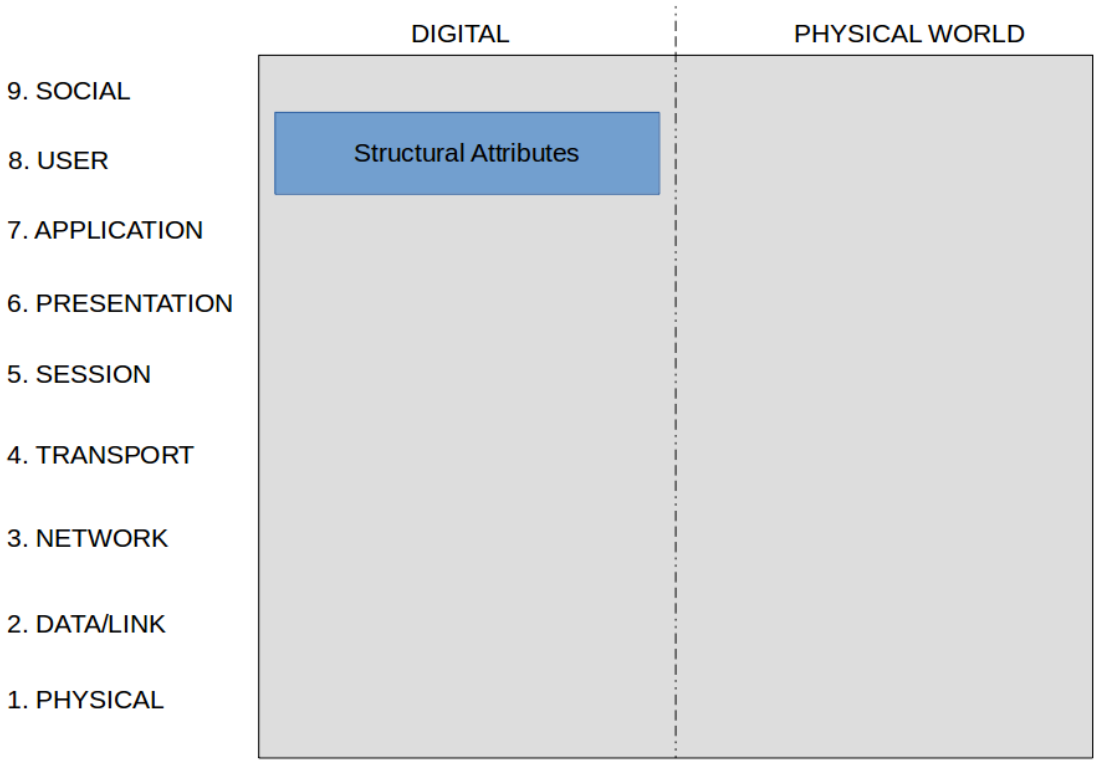

Fig. 4.11. Structural linguistic Library Coverage

Table 4.11: Structural attributes (8)

Structural attributes (8)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
6.8.100	Total number of lines	1	Integer	
6.8.101	Total number of sentences	1	Integer	
6.8.102	Total number of paragraphs	1	Integer	
6.8.103	Number of sentences per paragraph	1	Integer	
6.8.104	Number of characters per paragraph	1	Integer	
6.8.105	Number of words per paragraph	1	Integer	
6.8.106	Has separators between paragraphs	1	True/false	0 OR 1
6.8.110	E-Mail - Has quoted text	1	True/false	0 OR 1
6.8.111	E-Mail - position of quoted text	1	Text_String	Top, Bottom, Inline
6.8.112	E-Mail - Indentation of paragraph	1	True/false	0 OR 1
6.8.113	E-Mail - Use E-Mail as a signature	1	True/false	0 OR 1
6.8.114	E-Mail - Use Telephone as signature	1	True/false	0 OR 1
6.8.115	E-mail - Use URL as signature	1	True/false	0 OR 1
6.8.116	E-Mail - Has greeting	1	True/false	0 OR 1

4.5.8 Physical

The physical world attributes relate to the digital elements physical existence. An operating system exists within a physical computer, that has a real world location. That physical computer will be of a specific make and model, and will in itself contain several physical attributes. In the same way a networking piece of equipment will have physical characteristics such as make, model, MAC address and real world location. Finally all of these devices will be under the control of a user, who in a criminal investigation is the primary focus of the investigation. They will have a number of physical attributes (hair colour, eye colour, height etc) which are unlikely to assist in a digital investigation, as well a select number that will. For the purpose of an attribution library the second set have been focused on, but could be expanded to the first set if they are pertinent to an investigation. The physical world libraries that will be developed are:

- Identity
- Location
- Vehicle
- Network_Equipment
- Computer

4.5.9 Identity (10)

Identity (10)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
10.8.1	Surname	1		
10.8.2	Forename	1		
10.8.3	Nickname	1		
10.8.4	Age	1		
10.8.5	Date Of Birth	1		

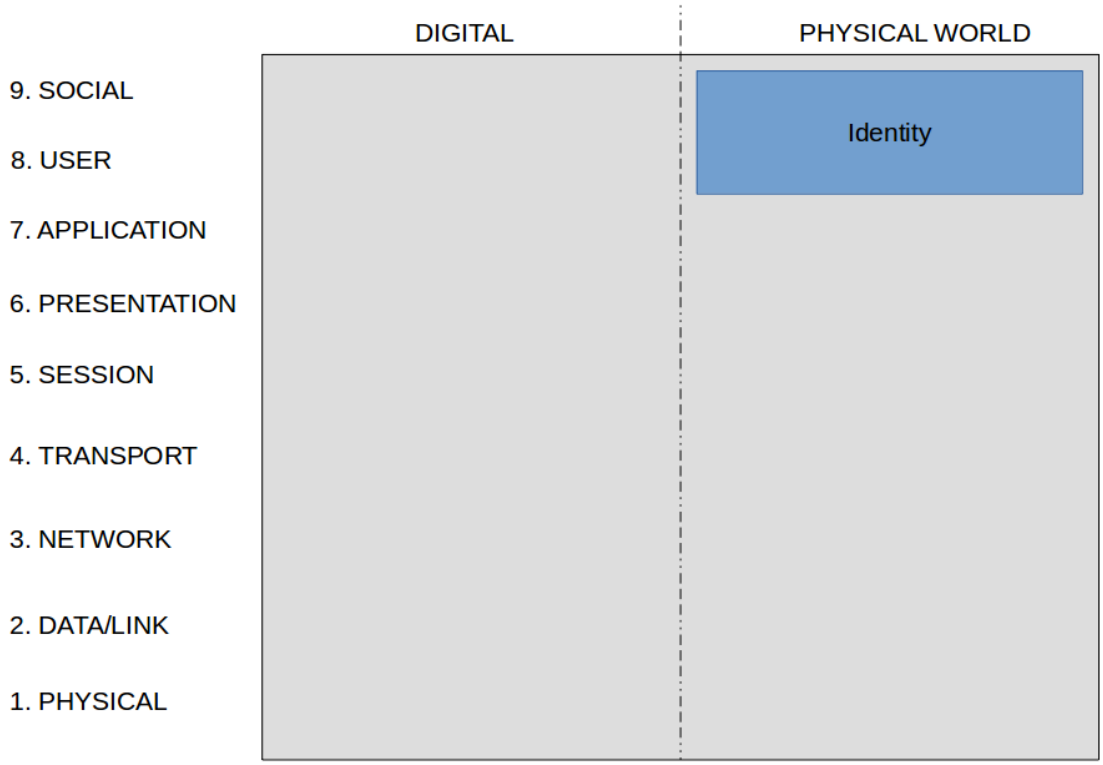

Fig. 4.12. Physical Identity Library Coverage

4.5.10 Location (11)

Location (11)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
11.1.1	House Number	1		
11.1.2	Street Name	1		
11.1.3	Town	1		
11.1.4	County	1		
11.1.5	Postcode	1		

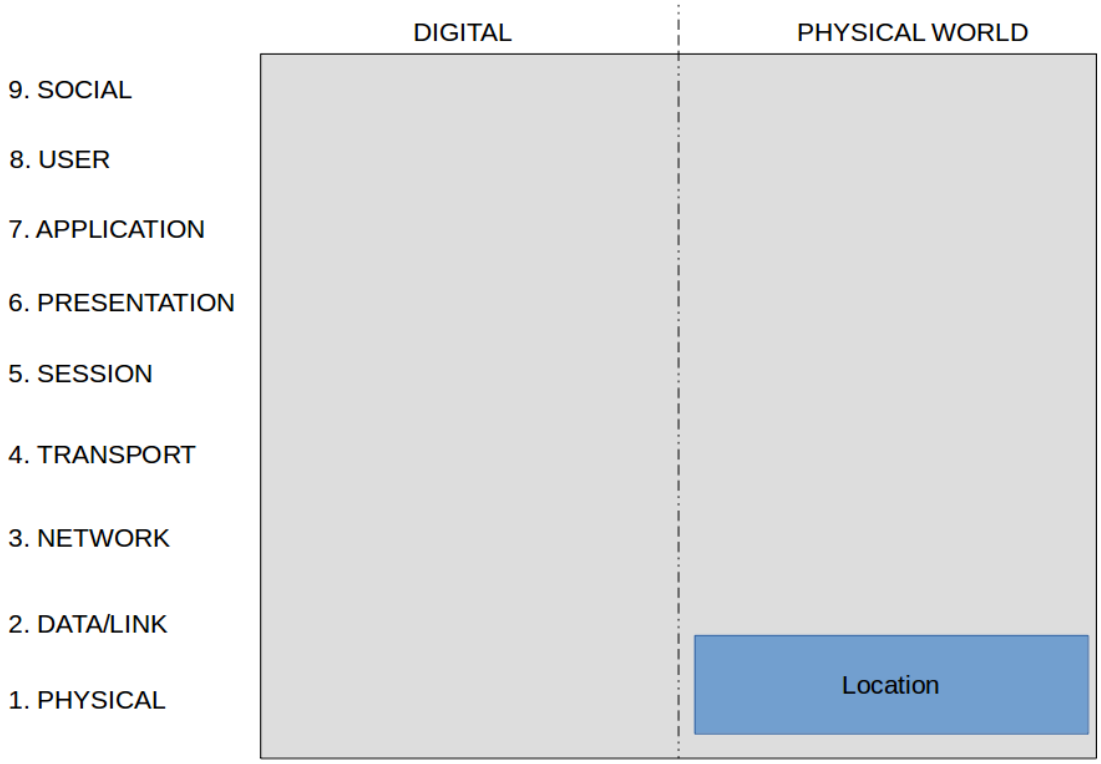

Fig. 4.13. Physical Location Library Coverage

4.5.11 Vehicle (12)

Vehicle (12)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
12.4.1	Make	1		
12.4.2	Model	1		
12.4.3	Colour	1		
12.4.4	Registration Number	1		
12.4.5	Fuel Type	1		

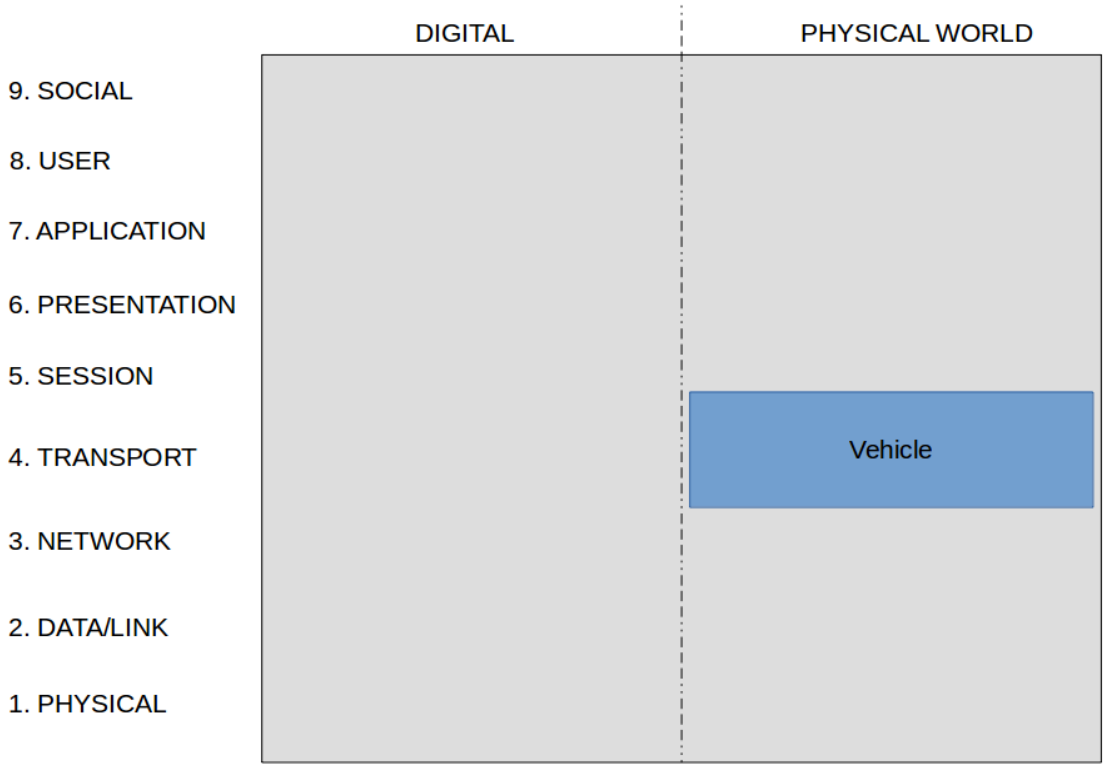

Fig. 4.14. Vehicle Library Coverage

4.5.12 Network_Equipment (13)

Network Equipment (13)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
13.1.1	Make	1		
13.1.2	Model	1		
13.1.3	Part Number	1		
13.1.4	Serial Number	1		
13.1.5	Firmware Version	1		
13.2.6	MAC address	1		

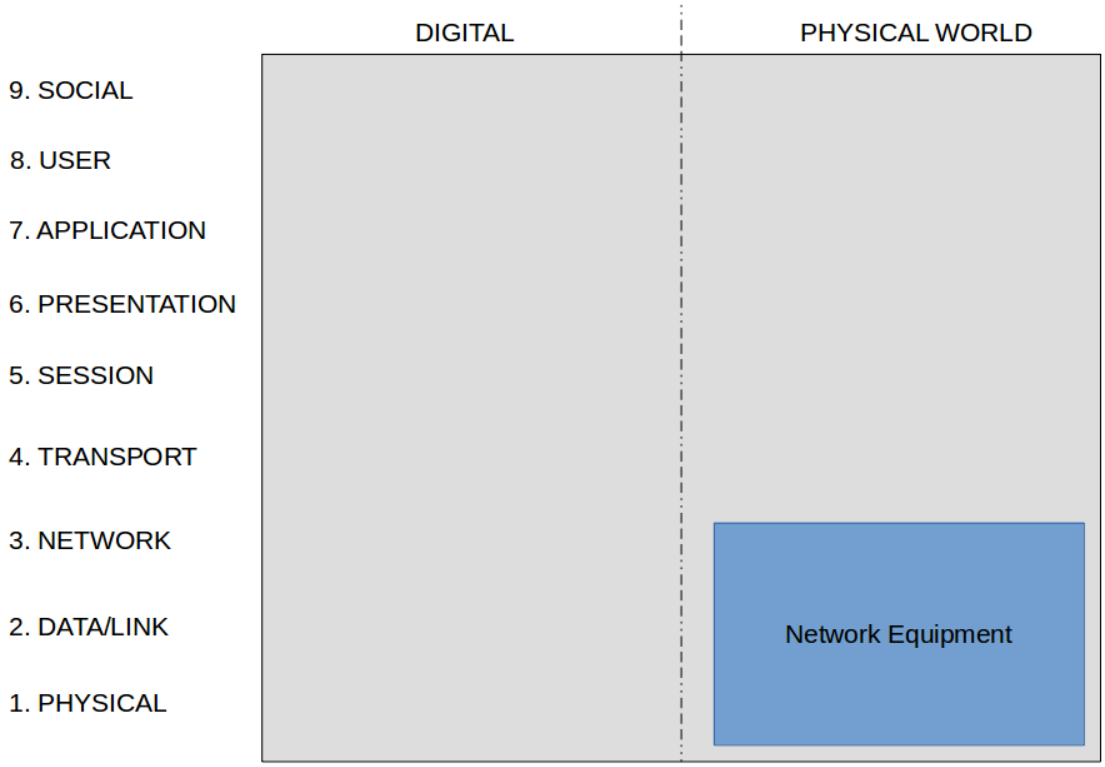

Fig. 4.15. Network Equipment Library Coverage

4.5.13 Computer (14)

Computer (14)				
UID	Attribute Name	Attribute Confidence	Attribute Type	Valid Attributes
14.1.1	Make	1		
14.1.2	Model	1		
14.1.3	Part Number	1		
14.1.4	Serial Number	1		
14.1.5	Firmware Version	1		
14.2.6	MAC address	1		
14.1.7	Graphics Card	1		
14.1.8	CPU speed	1		
14.1.9	Amount of RAM	1		

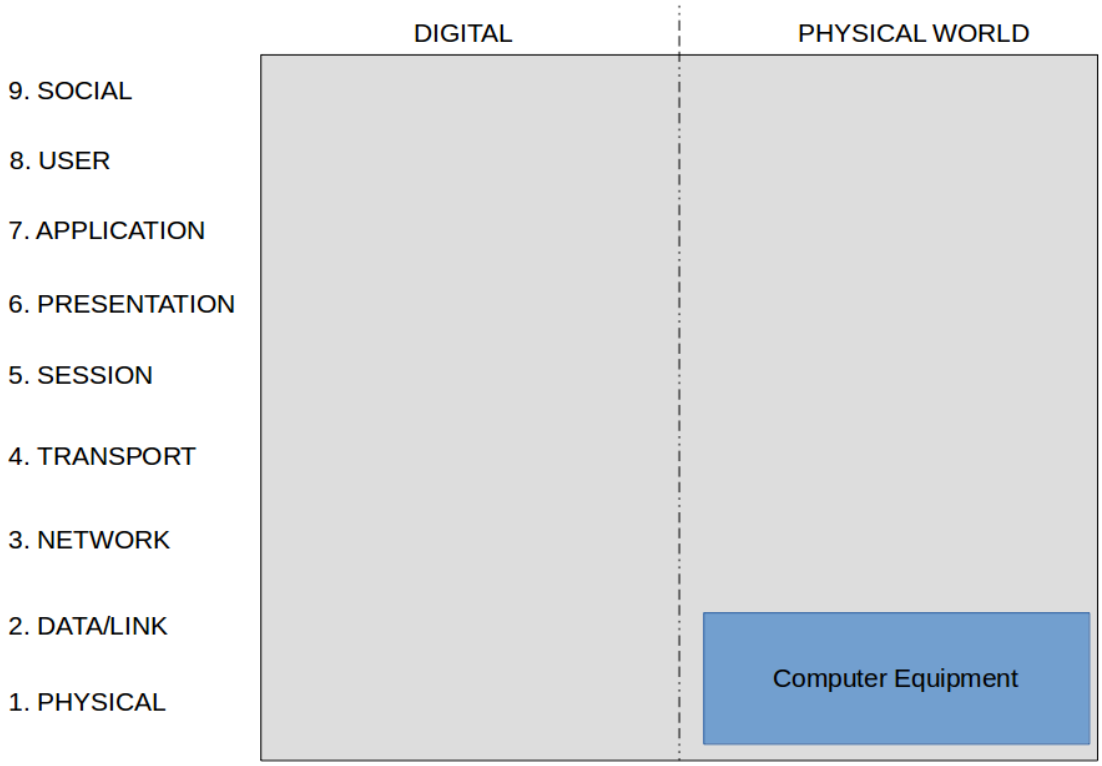

Fig. 4.16. Computer Equipment Library Coverage

4.6 Conclusions

A number of libraries have now been applied to the overarching model. As a result the model should be in a useable format that will enable testing. The method for developing a library has been demonstrated, enabling any expert to include their own library into the model, proving its extensibility. With all of the libraries overlaid it is possible to see immediately that there is good coverage of all areas of the model. The model as a whole will now need to be rigorously tested to prove its validity against a number of use cases.

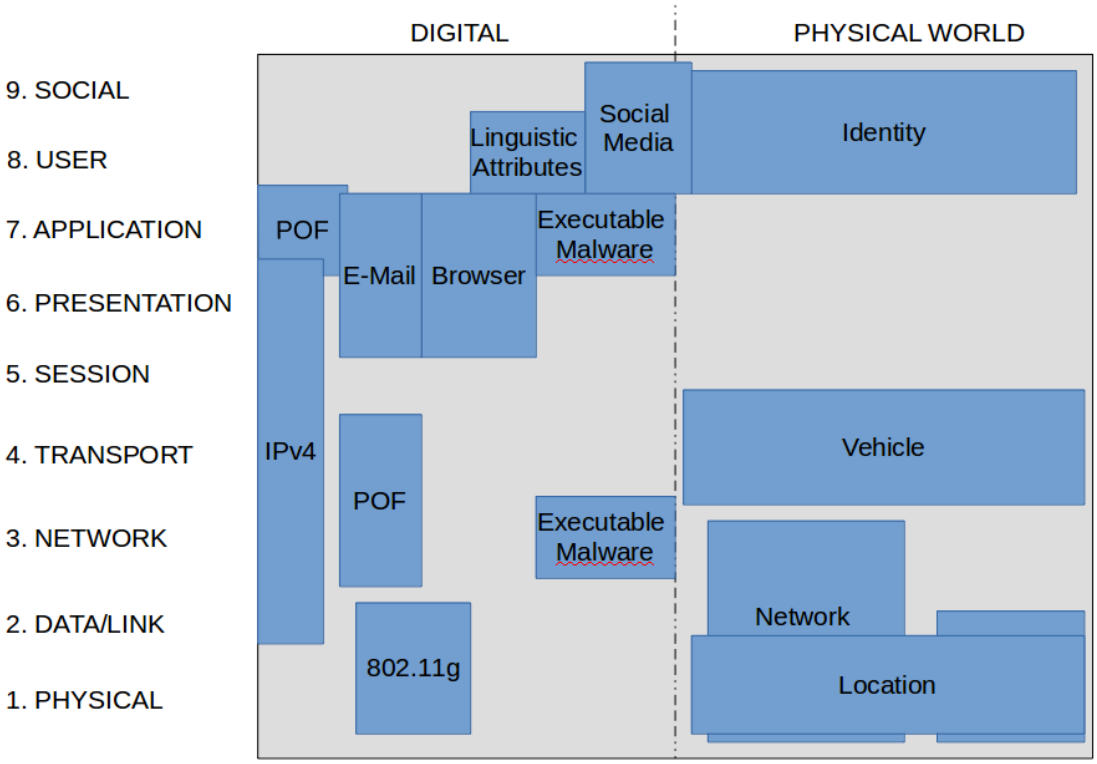

Fig. 4.17. Overall Library Coverage

Chapter 5

Experiments

5.1 Introduction

With a model having been developed, and a number of libraries produced to be used with the model there is now a requirement for the testing of both the model and a the libraries. It is foreseen that the model will be used in a number of use cases:

- As an analyst putting together a number of library attributes to ascertain a threat characteristics.
- A large data set is processed to highlight the significant attributes in a data set.
- The use of the framework in a real time realistic investigation.

For each of these use cases an experiment must be created to prove or disprove the models validity. In order to assess its usability two test groups were created and each candidate was presented with an attribution problem to solve. Each candidate used the framework to the best of their ability and gave feedback as to their experience. The first test group were primarily cyber security specialists, who have good knowledge of the technology involved in cyber attribution, but have not had any formal intelligence training. The second test group are trained intelligence analysts but lack any technical training. Each experiment was designed to test a different element of the initial framework aims.

5.2 Experiment 1

Experiment one is to assess the framework for its ability to highlight intelligence gaps. The candidates had to process the results of several different libraries, overlay them onto the framework and use the framework to identify any area that was lacking information. This first experiment resulted in a 100% success rate with all candidates successfully identifying the intelligence gap. Analysing the comments on the first task it was clear that the visualisation that the framework provides made the task trivially easy.

5.3 Experiment 2

Having identified an intelligence gap from the previous experiment, the candidates were now presented with a range of libraries and ask to select the library that would best resolve this gap. With the second experiment the results were split between the two groups. The cyber security specialists correctly identified the library that would produce the best results however only 72% of the intelligence analysts selected the correct library. Of the analysts who selected the incorrect library, they did select a library that would have covered the gap, but it was not suitable given the technology that was being used as identified from the initial libraries offered. This highlighted an important finding with regards to the use of the libraries. An elementary understanding of the technology is required to provide context for each library. When analysing deeper it was discover that the candidates who had incorrectly completed the task had already self identified as being technically weak in the initial candidate data capture. It is assessed that a minimal amount of training would be required to provide the context and enable these analysts to use the tool, and although anecdotally proven with the use of a 2 day course, no formal experimentation or assessment of requirements has been performed.

5.4 Experiment 3

Experiment three is to apply the skills learnt from the previous two experiments in order to complete a simulated attribution task. The candidates were provided once again with a series of library inputs. On this occasion they were given 60 minutes to process the inputs to the best of their ability and provide a complete attribution analysis. The result of the exercise was then captured to compare to the actual result as well as oral feedback captured to assess the difficulty the candidates experienced. In the experiment itself there were 3 protagonists who were lightly linked together and built of several independent libraries. There was a final person captured in the libraries with no link to the 3 protagonists. Although a time limit of 60 minutes was placed on the experiment, it was not expected that the candidates would reach this time limit. The results of the exercise showed some fascinating trends. Firstly it was found that the intelligence analysts in all but 2 cases finished before the cyber security analysts. When questioned about this fact it appears to be due to the intelligence analysts familiarity with the task. Asking an intelligence analyst to complete a link analysis is not an unusual request, no matter what the context of the information within those links. The cyber security analyst were able to complete the task to a good degree of success but having not completed a link analysis task before it appeared to take them longer to get going with the task. The oral feedback confirmed this with comments such as ”It took me a while to get going, but once I go the hang of it it made sense” and ”The hardest part was finding the links”. Despite the timing differences 94% of all candidates

correctly linked the 3 perpetrators in the scenario exercise. Of the candidates who failed to complete the link analysis, these were once again the intelligence analysts who had self identified as being technologically weak, and were missing some required context. Whilst the framework was successfully used to link the 3 perpetrators in the overall majority of cases a significant number of analysts incorrectly linked the 4th person to the group. 27% of the analysts found a link where there was none. When questioned about this there appeared to have been a misunderstanding in the task with the candidates believing that there had to be a link and they had made their best guess as to what it was. Whilst this is disappointing, it is also a significant problem in the real world with actions regularly being misattributed. Of the candidates who incorrectly made a connection 30% belonged to the intelligence analysts group and 70% belonged to the cyber security analysts. When analysing the links that had been made the intelligence analysts made a link at the lower more technical layers of the framework, where as the cyber analysts made the connection in the physical or social. This shows that the links were being made in the locations that the analyst is most outside of their comfort zone. If the experiment were to be repeated it would be made explicitly clear the potential that not all elements are connected.

5.5 Experiment 4

Experiment four was used to assess the frameworks capability to process large amounts of data and automatically perform statistical analysis. Should a framework be used for the creation of attribute data in a live environment, then it is highly likely that a very large data sets will be created in a relatively short space of time. Whilst this data set may well prove useless to an analyst without any specialist tools, if the data set can be processed in an automated fashion for quick wins, it may produce results which an analyst can immediately use. An example would be to process all captured attributes from a specific tool and analyse them to see which attributes reliably produce the best fingerprint or perform an automated link analysis based on a range of captured attributes that are known to be good indicators. It is critically important to know which attributes are the most significant in a data set. These will be the data points that produce the widest range of values (greatest entropy) whilst not being unique. In order to test this capability a small sample set was collected and analysed against the attributes held within the library. This created a population of critical mass which could then be used to assert key findings. These key findings were then verified against another test population. For the purpose of this test executable malware was analysed using a static analysis tool. The tool was run over a sample of 18,000 (N) previously known malware samples. The aim of the analysis is to ascertain which attributes which are extractable through static analysis may be useful in the attribution of an attacker when comparing against similar malware. The results of the static analysis found that there are a number of attributes that meet the requirements. These were:

- IP Address
- Domain
- Domain Generation Algorithm (DGA)
- E-Mail
- BitCoin Address
- Date of Compilation
- Dropped Files
- Packer
- Entropy
- Strings

It is common for malware to contain IP addresses. These are ordinarily locations to go for further payloads and/or command and control infrastructure. The development of a command and control infrastructure is a non trivial task and required considerable effort. As a result it is common for a malware developer to use the same C2 infrastructure for multiple variants of malware. In a similar context to IP addresses, Domain names may be used for command and control as well as payload collection points. The advantage of a domain name is that, should the IP address of the C2 become compromised, it is possible to fix the issue through the redirection of the domain to a new IP address, thus creating a healing malware network. A Domain Generation Algorithm (DGA) is a method of not directly encoding your domain as plain text into you malware, making it more difficult to detect. The DGA is seeded and will then come up with a seemingly random domain name. As long as the same seed is used, the same domains will be produced. This can also work on a time delay system, such that the seed will change every 12 hours to create another domain, making it extremely difficult to block. If you are able to isolate the DGA through static analysis, it was discovered that the same DGA was used in multiple samples and only the seed was changed. The data of compilation is based upon the the computer clock time. It was discovered that a number of malware developers have very inaccurate clock settings, often years out. That said, when families of malware were being released, the time between releases and the time between compilations often corresponded. The fact that the clocks were so far out made it easier to spot. BitCoins addresses are more regularly appearing in malware, primarily based around ransomware attacks. If the same bitcoin address is in use then it is the same attacker. Malware will often want to create persistence for itself, such that the malware will remain active even after the computer has rebooted. One of the primary methods for this is to create a file on the

infected computer end then call it during the boot sequence. This created file is called a dropped file. It was discovered that malware developer occasionally use the same name for their dropped files, or follow a pattern that is easy to distinguish. A packer process (sometimes referred to as a cryptex) is an attempt to evade antivirus by encrypting the malicous payload. The malware designer will put the malware through this packer process which will encrypt as much of the program as possible and interact with the encrypted blob. The result of this is that it is possible to ascertain the packer that has been used to create the malware. There are only a limited number of packers available, and an author is highly likely to pick one packer that they like and stick with it, especially as this is often a paid for service. The entropy of the file is highly linked to the packer that has been used. The entropy is the amount of apparent randomness found within a file. An ordinary file is likely to be made primarily of human readable characters. Because the malware has gone through an encryption process it mostly appears to be random gibberish. As a result a file with high entropy is almost certainly encrypted in some manor and could be malware. It is possible to pull out all human readable strings from any executable. Whilst the majority of these are nonsensical, it occasionally happens that you are able to extract passwords or other identifying strings. This is however a fairly unreliable source of information.

5.5.0.1 Verification of results

In order to verify the suspected attributes that are of use, the same attributes were used against known families of malware. These are malware that have gone beyond static analysis and have been fully reverse engineered by experts who have asserted that the malware is ”related”. The assumption is that related malware is likely to have the same author. This is a critical weakness, and a future experiment should be against a convicted criminals confirmed set of created malware, however it was not possible to acquire that data set. It was found that the same packer was used in every sample of the same family. This was not however the most reliable attribute, as there are only a limited number packers available, and the same packer was discovered across multiple families. Where IP addresses, Domains, and bitcoins were detected they were extremely reliable in proving linkage between differing malware. Unfortunately there were only a limited number of malware that could extract this data from a purely static analysis. This information could be extracted through other tools and means, such as a sandbox execution. The date of compilation was probably the most reliable method of linking malware samples based upon the release dates. This was reliant on knowing when a malware sample was first detected in the wild, and this external data may not always be available. Strings were the least reliable method, as there was simply too much noise. In one malware family a shared password was discovered, however this was the exception rather than the rule. If I was performing an analysis on two pieces of malware attempting to prove a link then I would look into the strings in both samples, but it would not be my first consideration, and would not scale easily. The

experiment verified the attributes selected from the original data set are the most prevalent for the use in attribution. The experiment however highlighted the requirement to not rely on an individual attribute in isolation, and it is the collection of attributes as a whole that create the fingerprint. This highlights the importance of the framework and the significant enhancement that having multidisciplinary attributes could bring to the attribution problem.

5.6 Experiment 5

The final experiment was based around the dissemination of captured information. The framework needs to be a suitable and reliable mechanism for the transmission of attribute data if is is to a useful intelligence tool. An opportunity arose to use the framework during a live demonstration of the dangers of over exposure on social media. For the demonstration the name of a BBC journalist and would be provided and as much open source intelligence on that journalist would have to be gathered whilst they were live on air. A colleague would be in the studio and feeding the information discovered to the presenter and needed to receive all collected information. For this purpose the model was used and my colleague had access to the site where the intelligence picture was developing. Due to only having 10 minutes prior warning it was not possible to teach the colleague the framework in any depth, so it would test how easy it was to extract the information. The experiment would run for a one hour period. Due to the nature of what was being asked the majority of the information would appear in layers 8 and 9 of the framework. Whilst the scope of the experiment was restricted to the single BBC presenter permission was given to broaden the search to discover more information as long as this information was not broadcast. All captured information was shared with the presenter in question and they were happy with the information that was released. Further information was captured and is included in the model although it has been obliterated to avoid unwanted disclosure. Despite this it is still possible to see what data was collected within the time frame. The framework proved to be a very flexible, although it was found at this point that there were not enough social media libraries pre built, and as a result it was required to develop and expand them whilst the experiment was running live. The fact that this is possible during a live experiment shows the power of having a framework, as it would have been very difficult to create an instantly usable product without a framework to place it upon. It was trivial to cluster data and create the links between the data clusters to create a more complete actor picture. Additional information was also placed into the model that would not ordinarily be captured as a part of an attribution investigation however was useful for the demonstration. The framework was flexible enough to cope with this. My colleague had no difficulty in extracting the information and the data remained in a structured format such that it was easy to extract the relevant information. The only comment was that because data was being added so quickly it would have been useful to have a method of highlighting new

information, or at least the latest information added through a simple highlighting method. The completed attribution picture is shown here:

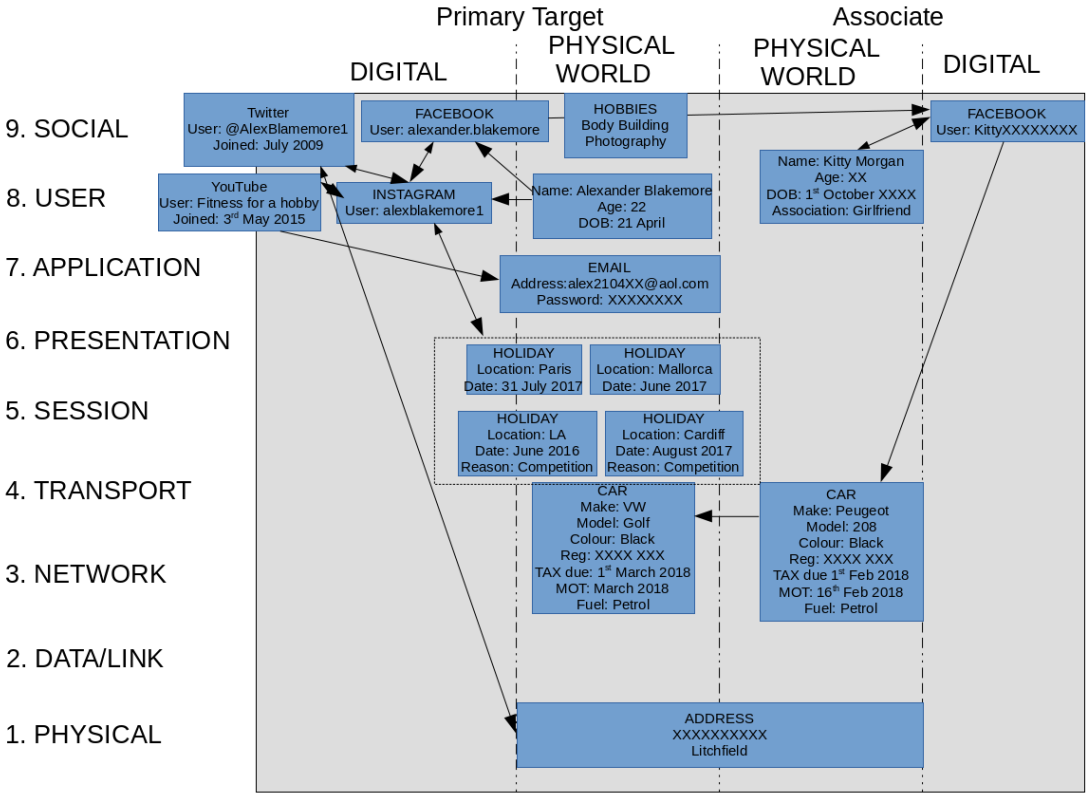

Fig. 5.1. Data captured during one hour OSINT exercise. Some data has been redacted by request of the owner.

This experiment proved conclusively that the framework is suitable for use in a live and fast paced investigation with the both the ease of creating data clusters and links between the clusters, as well as the ease of extracting the data in a live environment from a non trained analyst. Improvements have been suggest in the form a a method of highlighting newly added information. This will be taken forward as further work.

Chapter 6

Conclusions & Future Works

6.1 Conclusions

The objectives defined in the introduction were:

The framework should: Be suitable to accept any form of attribution data Be suitable to perform live attribution analysis Be capable of highlighting intelligence gaps Enable fast communications of attribution data, no matter what the source

At the conclusion of the project there is a proposed framework for the use of analysts working on the attribution problem. It was shown to be suitable for multidisciplinary work through the inclusion of linguistic and RF libraries. It is hoped that this framework will allow collaborative research in the future to improve attribution techniques. The procedural mechanism for the creation of new libraries enabled the rapid development of new libraries and will enable other experts or tools to enhance the current library set and enrich the attribution picture. The framework was used to disseminate in a real time environment during experiment 5. This was an especially pleasing demonstration of the framework as an information sharing platform as a colleague without any prior training was able to use the framework to extract the relevant information. The visual aspect of the framework was of real value during this task. This was further validated during experiments 2 and 3 in feedback from the analysts. Experiment 1 proved a complete success in identifying the intelligence gaps in an attribution task, although it did highlight that a there is an underlying knowledge requirement to successfully be able to apply the correct library to resolve the issue. This fundamental knowledge base is likely to expand as other specialist areas develop their libraries to include in the framework. There is a concern that with too many libraries the framework would become too cumbersome and a great level of knowledge would be required in order to use it. An alternative approach would be that of collaboration, with multiple analysts who have a range of specialisms. Although untested in a collaborative environment there are no known reasons as to why this would not work in practice, and further research would have to be performed. It is assessed that the objectives have been successfully met

although improvments are still possible. These will be expanded upon in the future works section.

6.1.1 Lessons Learned

Due to the ever changing landscape within the cyber world I found throughout the project that the development of methodologies was almost more important than the actual products themselves. This was especially the case with the development of libraries for use within the framework. A lot of time was wasted during the project developing libraries that ultimately were incompatible or out of date. This is why such a heavy emphasis in the write up is around the development process rather than specific libraries which could change. If completing the project again I would spend a considerable amount of time at the outset developing the methodology alongside a library, rather then developing the methodology almost in isolation having completed a number of libraries which ultimately had to be scraped. An extreme amount of time was spent on the learning of linguistic attribution techniques to see if these could directly be applied to the cyber realm. Ultimately this was a fruitless endeavour as it was discovered that the latest in linguistic attribution techniques is moving towards machine learning techniques that are already employed in the cyber security arena. Although this realisation occurred early the author continued to spend an extended amount of time continuing research in the field in the belief that the author was not understanding something critical. This lead to an unbalancing and unjustified amount of the time spent towards linguistic attribution which is reflected, although somewhat corrected for, in the final presentation of the project. The author should have trusted their instincts and the methodical approach to the literature review. It was also discovered that a large amount of literature on cyber attribution analysis is based around perfect systems of capture with no noise. A number of the proposed systems required a complete redesign of the fabric of the internet which is neither realistic nor useful in an actual environment. Whilst interesting proposals they hold no value to a real world systems and would fail against any realistic data set.

6.2 Future Works

From the experimentation it has been found that the framework could be improved through the highlighting of new information as it is being populated. This should be trivial to include and would enhance an analysts ability to brief directly from the tool.

Currently the experimentation has been performed on a very small scale, and whilst indications are very positive, it would be prudent to extend the experiments to a much larger sample size to prove statistical significance in a greater population.

The model has proven itself to be useful with only a limited set of libraries available for the testing of the framework. Obvious future development will be around the introduction of further libraries to enrich the

attribution picture even further. Proposed libraries would include, but are not limited to;

- HTML
- Scripting languages
- Social Media
- IPV6
- Bluetooth

HTML is used in the development of websites. Any website that is hosting illegal material must have been created by a web developer, and that developer will have their own unique style. It would be possible to capture some of these attributes through the analysis of the sites.

Scripting languages such and ruby, perl, bash, powershell and python would each require their own library. A scripting language has a far greater linguistic traits due to the fact that they have not gone through a compilation process which may obfuscate the original authors own style. Powershell, Python and Ruby should be prioritised due to their extensive use in cyber attacks.

An increase in the number of social media platforms that are included in the framework will be inevitable. This will develop quickly as social media platforms fall in and out of favour with the general public, as well as how quickly the applications themselves are updated providing new attributes or shutting down access to other attributes.

An obvious library that is missing is for IP Version 6. Whilst IPv6 has not been widely adopted yet it is only a matter of time, and as a result malicious activity will transfer to this new technology.Because it has not been widely adopted at the time of writing, it is not as well understood from an attribution perspective. There are some very exciting new features to enhance security, and a large variety of extra elements that are included by default within the IPv6 header. Each of these provides an opportunity as an attribute however the significance of each attribute is not yet understood.

A further library that may be of some limited use would be that of Bluetooth. It was highlighted during the creation of the framework that these is a general lack of published attribute research in the physical realm. Whilst this is highly likely to be due to the proximity negating the requirement for in depth technical research into attributes there are a number of applications for this in real world investigations. Bluetooth would be an obvious first candidate within the physical realm due to its promiscuity in modern technology.

The attribution framework itself is able to stand alone as an intelligence product, however it would be of more value if integrated with other models to assist an intelligence analyst in being able to provide valid and actionable intelligence. As an example this framework would be greatly assisted by the Diamond

threat modelling methodology of intrusion analysis Caltagrone et al. (2013). The archetype threat actor framework would be greatly enhanced through the application of the cyber kill chain Martin (2014) in order to provide the indicators of compromise (IOC'S) expected from each type of archetypal actor at each stage of the cyber kill chain. Not only would this map the IOC's enabling the quick assessment of the threat but also this could then be used as a predictive tool to assist an analyst for expected next actions. Finally to increase the adoption and frameworks interoperability in would make logical sense to adopt a widely recognised data exchange format such as STIX Barnum (2012). This is specifically designed to enable cyber threat intelligence sharing between different platforms. Additional work is required to develop a method of using STIX to transmit the data captured in this framework.

7

Reference

[1] Information assurance standard 1, 2009.

[2] Joint service publication 440 (restricted), 2009.

[3] 802.11-2012 - IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Technical Report IEEE Std 802.11-2012, IEEE-Inst, 2012.

[4] A. Abbasi and H. Chen. Visualizing authorship for identification. *Intelligence and Security Informatics*, pages 60–71, 2006.

[5] Lada A Adamic and Bernardo A Huberman. Zipfs law and the internet. *Glottometrics*, 3(1):143–150, 2002.

[6] Micah Adler. Trade-offs in probabilistic packet marking for ip traceback. *Journal of the ACM (JACM)*, 52(2):217–244, 2005.

[7] Micah Adler, Jeff Edmonds, and Jivri Matousek. Towards asymptotic optimality in probabilistic packet marking. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 450–459. ACM, 2005.

[8] Mohammed Alenezi and Martin J Reed. Efficient as dos traceback. In *Computer Applications Technology (ICCAT), 2013 International Conference on*, pages 1–5. IEEE, 2013.

[9] Hassan Aljifri. Ip traceback: a new denial-of-service deterrent? *IEEE Security & Privacy*, 99(3):24–31, 2003.

[10] Armen E Allahverdyan, Weibing Deng, and Qiuping A Wang. Explaining zipf's law via a mental lexicon. *Physical Review E*, 88(6):062804, 2013.

[11] Gabriel Altmann. Zipfian linguistics. *Glottometrics*, 3:19–26, 2002.

[12] Ion Androutsopoulos, John Koutsias, Konstantinos V Chandrinos, George Paliouras, and Constantine D Spyropoulos. An evaluation of naive bayesian anti-spam filtering. *arXiv preprint cs/0006013*, 2000.

[13] Ion Androutsopoulos, John Koutsias, Konstantinos V Chandrinos, and Constantine D Spyropoulos. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 160–167. ACM, 2000.

[14] Mark Aronoff and Janie Rees-Miller. *The handbook of linguistics*, volume 43. John Wiley & Sons, 2003.

[15] Randall Atkinson and Stephen Kent. Ip authentication header. 1998.

[16] Eric Backer and Peter van Kranenburg. On musical stylometrya pattern recognition approach. *Pattern Recognition Letters*, 26(3):299–309, 2005.

[17] Joan Beal. *Language and region*. Taylor & Francis, 2006.

[18] A. Belenky and N. Ansari. On ip traceback. *Communications Magazine, IEEE*, 41(7):142–153, 2003.

[19] S.M. Bellovin, M. Leech, and T. Taylor. Icmp traceback messages. Technical report, 2000.

[20] Douglas Biber, Susan Conrad, and Randi Reppen. *Corpus linguistics: Investigating language structure and use*. Cambridge University Press, 1998.

[21] Barry J Blake. *All About Language: A Guide*. Oxford University Press, 2008.

[22] Avrim Blum, Dawn Song, and Shobha Venkataraman. Detection of interactive stepping stones: Algorithms and confidence bounds. In *International Workshop on Recent Advances in Intrusion Detection*, pages 258–277. Springer, 2004.

[23] Danah Boyd, Scott Golder, and Gilad Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In *System Sciences (HICSS), 2010 43rd Hawaii International Conference on*, pages 1–10. IEEE, 2010.

[24] Aviva Briefel. *The deceivers: Art forgery and identity in the nineteenth century*. Cornell University Press, 2006.

[25] Cameron SD Brown. Cyber-attacks, retaliation and risk: Legal and technical implications for. *Cybersecurity Policies and Strategies for Cyberwarfare Prevention*, page 166, 2015.

[26] John F Burrows. Numbering the streaks of the tulip? reflections on a challenge to the use of statistical methods in computational stylistics. *CH Working Papers*, 1(1), 2005.

[27] S. Burrows and SMM Tahaghoghi. Source code authorship attribution using n-grams. In *Proceedings of the Twelth Australasian Document Computing Symposium, Melbourne, Australia, RMIT University*, pages 32–39. Citeseer, 2007.

[28] Lyle Campbell. *Historical linguistics: An introduction*. MIT press, 1998.

[29] David Canter and Joanne Chester. Investigation into the claim of weighted cusum in authorship attribution studies. *International Journal of Speech Language and the Law*, 4(2):252–261, 1997.

[30] Nicholas B Chang and Mingyan Liu. Controlled flooding search in a large network. *IEEE/ACM Transactions on Networking*, 15(2):436–449, 2007.

[31] Bo-Chao Cheng, Guo-Tan Liao, Ching-Kai Lin, Shih-Chun Hsu, Ping-Hai Hsu, and Jong Hyuk Park. Mib-itrace-cp: An improvement of icmp-based traceback efficiency in network forensic analysis. In *IFIP International Conference on Network and Parallel Computing*, pages 101–109. Springer, 2012.

[32] Mi-Jung Choi, James W Hong, and Hong-Taek Ju. Xml-based network management for ip networks. *ETRI journal*, 25(6):445–463, 2003.

[33] Noam Chomsky. *Aspects of the Theory of Syntax*. Number 11. MIT press, 1965.

[34] Noam Chomsky. Conditions on rules of grammar. *Linguistic analysis*, 2(4):303–351, 1976.

[35] Noam Chomsky and Robert DiNozzi. *Language and mind*. Harcourt Brace Jovanovich New York, 1972.

[36] John Ellery Clark, Colin Yallop, and Janet Fletcher. An introduction to phonetics and phonology. 1995.

[37] Paul Clough. Plagiarism in natural and programming languages: an overview of current tools and technologies, 2000.

[38] Bernard Comrie. *Language universals and linguistic typology: Syntax and morphology*. University of Chicago press, 1989.

[39] Alicia Skinner Cook, Janet J Fritz, Barbara L McCornack, and Cris Visperas. Early gender differences in the functional usage of language. *Sex Roles*, 12(9-10):909–915, 1985.

[40] Allan Cook, Andrew Nicholson, Helge Janicke, Leandros A Maglaras, and Richard Smith. Attribution of cyber attacks on industrial control systems. *EAI Endorsed Trans. Indust. Netw. & Intellig. Syst.*, 3(7):e3, 2016.

[41] Vivian James Cook and Mark Newson. *Chomsky's universal grammar: An introduction*. Blackwell Oxford, 1988.

[42] Florian Coulmas. *Sociolinguistics: The study of speakers' choices*. Cambridge University Press, 2005.

[43] M. Coulthard and A. Johnson. *An introduction to forensic linguistics: language in evidence*. Psychology Press, 2007.

[44] Daryl C Cromer, David B Rhoades, Howard J Locker, James P Ward, Eric R Kern, Brandon J Ellison, and Richard A Dayan. Method and system for providing protection against theft and loss of a portable computer system, October 11 2005. US Patent 6,954,147.

[45] Ferdinand De Saussure. Nature of the linguistic sign. *Course in general linguistics*, 1916.

[46] O. De Vel, A.M. Anderson, M.W. Corney, and G.M. Mohay. E-mail authorship attribution for computer forensics. *Applications of Data Mining in Computer Security*, 6:–, 2002.

[47] D. Dean, M. Franklin, and A. Stubblefield. An algebraic approach to ip traceback. *ACM Transactions on Information and System Security (TISSEC)*, 5(2):119–137, 2002.

[48] Collins English Dictionary. Collins. *UK: Harper Collins Publishers*, 1991.

[49] Oxford English Dictionary. Oxford english dictionary, 2003.

[50] Dictionary.com. Dictionary.com. 2017.

[51] Harris Drucker, S Wu, and Vladimir N Vapnik. Support vector machines for spam categorization. *Neural Networks, IEEE Transactions on*, 10(5):1048–1054, 1999.

[52] P. Eckersley. How unique is your web browser? In *Privacy Enhancing Technologies*, pages 1–18. Springer, 2010.

[53] Alvar Ellegård. *A Statistical method for determining authorship: the Junius Letters, 1769-1772*, volume 13. Göteborg: Acta Universitatis Gothoburgensis, 1962.

[54] Jillian M Farringdon, Andrew Queen Morton, Michael G Farringdon, and M David Baker. *Analysing for Authorship: A Guide to the Cusum Technique*. University of Wales Press Cardiff, 1996.

[55] Gary Alan Fine. Cheating history: The rhetorics of art forgery. *Empirical Studies of the Arts*, 1(1):75–93, 1983.

[56] Stuart James Fleming. *Authenticity in art. The scientific detection of forgery*. Institute of Physics; Crane, Rusack and Co., 1975.

[57] Daniel Flemming and Neil Rowe. Cyber coercion: Cyber operations short of cyberwar. In *Proceedings of the 10th International Conference on Cyberwarfare and Security ICCWS-2015, Skukuza, South Africa, March*, pages 95–101, 2015.

[58] Simona Florescu, Christine Körner, Michael Mock, and Michael May. Efficient mobility pattern stream matching on mobile devices. In *Proc. of the Ubiquitous Data Mining Workshop (UDM 2012)*, pages 23–27, 2012.

[59] Donald W Foster. Primary culprit: An analysis of a novel of politics. *New York (26 February 1996)*, 50:57, 1996.

[60] Roger Fowler. On critical linguistics1. *Texts and practices: Readings in critical discourse analysis*, page 1, 1996.

[61] G. Frantzeskou, S. Gritzalis, and S. MacDonell. Source code authorship analysis for supporting the cybercrime investigation process. In *1st International Conference on eBusiness and Telecommunication Networks-Security and Reliability in Information Systems and Networks Track, Setubal Portugal*, pages –. Citeseer, 2004.

[62] G. Frantzeskou, E. Stamatatos, S. Gritzalis, C.E. Chaski, and BS Howald. Identifying authorship by byte-level n-grams: The source code author profile (scap) method. *Int. Journal of Digital Evidence*, 6(1):–, 2007.

[63] G. Frantzeskou, E. Stamatatos, S. Gritzalis, and S. Katsikas. Effective identification of source code authors using byte-level information. In *Proceedings of the 28th international conference on Software engineering*, pages 893–896. ACM, 2006.

[64] Glenn Fung. The disputed federalist papers: Svm feature selection via concave minimization. In *Proceedings of the 2003 Conference on Diversity in Computing*, pages 42–46. ACM, 2003.

[65] Anthony D Glosson. *Active Defense: An Overview of the Debate and a Way Forward*. George Mason University, Mercatus Center, 2015.

[66] Joshua T Goodman. A bit of progress in language modeling. *Computer Speech & Language*, 15(4):403–434, 2001.

[67] Michael T Goodrich. Probabilistic packet marking for large-scale ip traceback. *IEEE/ACM Transactions on networking*, 16(1):15–24, 2008.

[68] J. Grieve. Quantitative authorship attribution: An evaluation of techniques. *Literary and Linguistic Computing*, 22(3):251–, 2007.

[69] G. Gupta, C. Mazumdar, and MS Rao. Digital forensic analysis of e-mails: A trusted e-mail protocol. *International Journal of Digital Evidance*, 2(4):–, 2004.

[70] RA Hardcastle. Cusum: A credible method for the determination of authorship? *Science & Justice*, 37(2):129–138, 1997.

[71] Margaret Harris. *Language experience and early language development: From input to uptake*. Psychology Press, 2013.

[72] Randy Allen Harris. *The linguistics wars*. Oxford University Press, 1993.

[73] J.H. Hayes. Authorship attribution: A principal component and linear discriminant analysis of the consistent programmer hypothesis. *International Journal of Computers and their Applications (IJCA)*, 15(2):79–99, 2008.

[74] Mario Heiderich, Marcus Niemietz, Felix Schuster, Thorsten Holz, and Jörg Schwenk. Scriptless attacks: stealing the pie without touching the sill. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 760–771. ACM, 2012.

[75] Mary P Hiatt. *The way women write*. Teachers College Press New York, 1977.

[76] Michael L Hilton and David I Holmes. An assessment of cumulative sum charts for authorship attribution. *Literary and Linguistic Computing*, 8(2):73–80, 1993.

[77] David I Holmes. The analysis of literary style–a review. *Journal of the Royal Statistical Society. Series A (General)*, pages 328–341, 1985.

[78] David I Holmes and Richard S Forsyth. The federalist revisited: New directions in authorship attribution. *Literary and Linguistic Computing*, 10(2):111–127, 1995.

[79] David I Holmes, Lesley J Gordon, and Christine Wilson. A widow and her soldier: Stylometry and the american civil war. *Literary and Linguistic Computing*, 16(4):403–420, 2001.

[80] David I Holmes and Fiona J Tweedie. Forensic stylometry: A review of the cusum controversy. *Revue Informatique et Statistique dans les Sciences Humaines*, 31:19–47, 1995.

[81] David L Hoover. Another perspective on vocabulary richness. *Computers and the Humanities*, 37(2):151–178, 2003.

[82] Luděk Hřebíček. Zipfs law and text. *Glottometrics*, page 27, 2002.

[83] J. Hunker, B. Hutchinson, and J. Margulies. Role and challenges for sufficient cyberattack attribution. *Dartmouth College: The Institute for Information Infrastructure Protection (The I3P)*, 28:–, 2008.

[84] Earl Hunt and Franca Agnoli. The whorfian hypothesis: A cognitive psychology perspective. *Psychological Review*, 98(3):377, 1991.

[85] Yannis M Ioannides and Henry G Overman. Zipfs law for cities: an empirical examination. *Regional science and urban economics*, 33(2):127–137, 2003.

[86] John Ioannidis. Ipsec. *Encyclopedia of Cryptography and Security*, pages 635–638, 2011.

[87] Vikas Jayaswal, William Yurcik, and David Doss. Internet hack back: Counter attacks as self-defense or vigilantism? In *Technology and Society, 2002.(ISTAS'02). 2002 International Symposium on*, pages 380–386. IEEE, 2002.

[88] Neil F Johnson, Zoran Duric, and Sushil Jajodia. *Information Hiding: Steganography and Watermarking-Attacks and Countermeasures: Steganography and Watermarking: Attacks and Countermeasures*, volume 1. Springer Science & Business Media, 2001.

[89] Abhishek Joshi and Rayan H Goudar. The attack back mechanism: An efficient back-hacking technique. In *Advanced Computing, Networking and Informatics-Volume 2*, pages 233–240. Springer, 2014.

[90] P. Juola. Authorship attribution. *Foundations and Trends in information Retrieval*, 1(3):233–334, 2006.

[91] P. Juola and R.H. Baayen. A controlled-corpus experiment in authorship identification by cross-entropy. *Literary and Linguistic Computing*, 20(Suppl):59–, 2005.

[92] P. Juola and J. Sofko. Proving and improving authorship attribution technologies. In *Proceedings of Canadian Symposium for Text Analysis (CaSTA)*, pages –. Citeseer, 2004.

[93] P. Juola, J. Sofko, and P. Brennan. A prototype for authorship attribution studies. *Literary and Linguistic Computing*, 21(2):169–, 2006.

[94] Slava Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 35(3):400–401, 1987.

[95] L.G. Kersta. Voiceprint identification. *Nature*, 196(4861):1253–1257, 1962. cited By (since 1996)45.

[96] Vlado Kešelj, Fuchun Peng, Nick Cercone, and Calvin Thomas. N-gram-based author profiles for authorship attribution. In *Proceedings of the conference pacific association for computational linguistics, PACLING*, volume 3, pages 255–264, 2003.

[97] Bradley Kjell. Authorship determination using letter pair frequency features with neural network classifiers. *Literary and Linguistic Computing*, 9(2):119–124, 1994.

[98] Christian Konrad and Frédéric Magniez. Validating xml documents in the streaming model with external memory. *ACM Transactions on Database Systems (TODS)*, 38(4):27, 2013.

[99] M. Koppel and J. Schler. Exploiting stylistic idiosyncrasies for authorship attribution. In *Proceedings of IJCAI*, volume 3, pages 69–72. Citeseer, 2003.

[100] M. Koppel, J. Schler, and S. Argamon. Computational methods in authorship attribution. *Journal of the American Society for information Science and Technology*, 60(1):9–26, 2009.

[101] M. Koppel, J. Schler, and S. Argamon. Authorship attribution in the wild. *Language Resources and Evaluation*, pages 1–12, 2011.

[102] M. Koppel, J. Schler, S. Argamon, and E. Messeri. Authorship attribution with thousands of candidate. pages –, 2006.

[103] Turgay Korkmaz, Chao Gong, Kamil Sarac, and Sandra G Dykes. Single packet ip traceback in as-level partial deployment scenario. *International Journal of Security and Networks*, 2(1-2):95–108, 2007.

[104] I. Krsul and E.H. Spafford. Authorship analysis: Identifying the author of a program. *Computers & Security*, 16(3):233–257, 1997.

[105] Christopher Krügel, Thomas Toth, and Engin Kirda. Service specific anomaly detection for network intrusion detection. In *Proceedings of the 2002 ACM symposium on Applied computing*, pages 201–208. ACM, 2002.

[106] Louis Kruh. A basic probe of the beale cipher as a bamboozlement. *Cryptologia*, 6(4):378–382, 1982.

[107] Patricia K Kuhl, Jean E Andruski, Inna A Chistovich, Ludmilla A Chistovich, Elena V Kozhevnikova, Viktoria L Ryskina, Elvira I Stolyarova, Ulla Sundberg, and Francisco Lacerda. Cross-language analysis of phonetic units in language addressed to infants. *Science*, 277(5326):684–686, 1997.

[108] Patricia K Kuhl, Karen A Williams, Francisco Lacerda, Kenneth N Stevens, and Björn Lindblom. Linguistic experience alters phonetic perception in infants by 6 months of age. *Science*, 255(5044):606–608, 1992.

[109] Roger Lass. *Historical linguistics and language change*, volume 81. Cambridge University Press, 1997.

[110] R. Layton, P. Watters, and R. Dazeley. Authorship attribution for twitter in 140 characters or less. In *2010 Second Cybercrime and Trustworthy Computing Workshop*, pages 1–8. IEEE, 2010.

[111] Aleksandar Lazarevic, Levent Ertoz, Vipin Kumar, Aysel Ozgur, and Jaideep Srivastava. A comparative study of anomaly detection schemes in network intrusion detection. In *Proceedings of the 2003 SIAM International Conference on Data Mining*, pages 25–36. SIAM, 2003.

[112] David D Lewis. Feature selection and feature extraction for text categorization. In *Proceedings of the workshop on Speech and Natural Language*, pages 212–217. Association for Computational Linguistics, 1992.

[113] James Andrew Lewis. Aux armes, citoyens: Cyber security and regulation in the united states. *Telecommunications Policy*, 29(11):821–830, 2005.

[114] Hwei-Jen Lin, Chun-Wei Wang, Yang-Ta Kao, et al. Fast copy-move forgery detection. *WSEAS Transactions on Signal Processing*, 5(5):188–197, 2009.

[115] Zhiqiang Lin. Automating introspection and forensics software development via binary code reuse. 2014.

[116] Jenshiub Liu, Zhi-Jian Lee, and Yeh-Ching Chung. Efficient dynamic probabilistic packet marking for ip traceback. In *Networks, 2003. ICON2003. The 11th IEEE International Conference on*, pages 475–480. IEEE, 2003.

[117] Yali Liu, Cherita Corbett, Ken Chiang, Rennie Archibald, Biswanath Mukherjee, and Dipak Ghosal. Sidd: A framework for detecting sensitive data exfiltration by an insider attack. In *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on*, pages 1–10. IEEE, 2009.

[118] Harold Love. *Attributing authorship: An introduction*. Cambridge University Press, 2002.

[119] Siwei Lyu, Daniel Rockmore, and Hany Farid. A digital technique for art authentication. *Proceedings of the National Academy of Sciences of the United States of America*, 101(49):17006–17010, 2004.

[120] Graham Mallinson and Barry J Blake. *Language typology: Cross-linguistic studies in syntax*, volume 46. North-Holland Amsterdam, 1981.

[121] Allison Mankin, Daniel Massey, Chien-Lung Wu, Shyhtsun Felix Wu, and Lixia Zhang. On design and evaluation of" intention-driven" icmp traceback. In *Computer Communications and Networks, 2001. Proceedings. Tenth International Conference on*, pages 159–165. IEEE, 2001.

[122] Ian Mann. *Hacking the human: social engineering techniques and security countermeasures*. Gower Publishing, Ltd., 2010.

[123] William B McGregor. *Linguistics: An Introduction*. Continuum, 2009.

[124] Mark Meiss, John Duncan, Bruno Gonçalves, José J Ramasco, and Filippo Menczer. What's in a session: tracking individual behavior on the web. In *Proceedings of the 20th ACM conference on Hypertext and hypermedia*, pages 173–182. ACM, 2009.

[125] Jan E Messerschmidt. Hackback: Permitting retaliatory hacking by non-state actors as proportionate countermeasures to transboundary cyberharm. *Colum. J. Transnat'l L.*, 52:275, 2013.

[126] Geoffrey Stewart Morrison. Forensic voice comparison and the paradigm shift. *Science & Justice*, 49(4):298 – 308, 2009.

[127] Andrew Q Morton. Once. a test of authorship based on words which are not repeated in the sample. *Literary and Linguistic Computing*, 1(1):1–8, 1986.

[128] Andrew Queen Morton and Sidney Michaelson. *The qsum plot*, volume 3. University of Edinburgh, Department of Computer Science, 1990.

[129] Frederick Mosteller and David Wallace. Inference and disputed authorship: The federalist. 1964.

[130] Biswanath Mukherjee, L Todd Heberlein, and Karl N Levitt. Network intrusion detection. *IEEE network*, 8(3):26–41, 1994.

[131] Christopher J Murphy. *Healthcare Industry Held Hostage: Cyberattacks and the Effect on Healthcare Critical Infrastructure*. PhD thesis, Utica College, 2017.

[132] Andrew Y Ng, Alice X Zheng, and Michael I Jordan. Stable algorithms for link analysis. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 258–266. ACM, 2001.

[133] Andrew Nicholson. Wide spectrum attribution: Using deception for attribution intelligence in cyber attacks. 2015.

[134] Andrew Nicholson, Tim Watson, Peter Norris, Alistair Duffy, and Roy Isbell. A taxonomy of technical attribution techniques for cyber attacks. In *European Conference on Cyber Warfare and Security*, page 188. Academic Conferences International Limited, 2012.

[135] S. Northcutt and J. Novak. *Network intrusion detection: An analyst's handbook*. New Riders Publishing, 2002.

[136] J. Olsson. *Forensic linguistics*. Continuum, 2008.

[137] Rebecca L Oxford. *Language learning styles and strategies*. Mouton de Gruyter, 2003.

[138] Kihong Park and Heejo Lee. On the effectiveness of probabilistic packet marking for ip traceback under denial of service attack. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 338–347. IEEE, 2001.

[139] Lee Pederson and Frederic G Cassidy. Chicago words: The regional vocabulary. *American Speech*, 46(3/4):163–192, 1971.

[140] F. Peng, D. Schuurmans, V. Keselj, and S. Wang. Automated authorship attribution with character level language models. In *10th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2003)*, pages 267–274, 2003.

[141] Roger D Peng and Nicolas W Hengartner. Quantitative analysis of literary styles. *The American Statistician*, 56(3):175–185, 2002.

[142] Robert Penhallurick. Welsh english: phonology. *A Handbook of Varieties of English*, 1:98–112, 2004.

[143] Gungor Polatkan, Sina Jafarpour, Andrei Brasoveanu, Shannon Hughes, and Ingrid Daubechies. Detection of forgery in paintings using supervised learning. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 2921–2924. IEEE, 2009.

[144] Stanislav Ponomarev, Nathan Wallace, and Travis Atkison. Detection of ssh host spoofing in control systems through network telemetry analysis. In *Proceedings of the 9th Annual Cyber and Information Security Research Conference*, pages 21–24. ACM, 2014.

[145] Tony Proctor. The development of cyber security warning, advice and report points. In *Nordic Conference on Secure IT Systems*, pages 61–72. Springer, 2012.

[146] Mahmoud T Qassrawi and Zhang Hongli. Deception methodology in virtual honeypots. In *Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference on*, volume 2, pages 462–467. IEEE, 2010.

[147] R Smyth R. Smith and B Hallaq. Fast: Framework for the assesment and mitigation of satellite threat. 2015.

[148] Wei Ren and Hai Jin. Honeynet based distributed adaptive network forensics and active real time investigation. In *Proceedings of the 2005 ACM symposium on Applied computing*, pages 302–303. ACM, 2005.

[149] Phil Rose. *Forensic speaker identification*. CRC Press, 2003.

[150] Ronald Rousseau. George kingsley zipf: Life, idea, his law and informetrics. *Glottometrics*, 3:11–18, 2002.

[151] J. Rudman. The state of authorship attribution studies: Some problems and solutions. *Computers and the Humanities*, 31(4):351–365, 1997.

[152] J. Rudman. Dna and nontraditional authorship attribution: An inclusive model. ALLC 2002 Tubingen, Germany, 2002.

[153] J. Rudman. Cherry picking in nontraditional authorship attribution studies. *CHANCE-BERLIN THEN NEW YORK-*, 16(2):26–32, 2003.

[154] Hargevik S. Stieg hargevik: The disputed assignment of memoirs of an english officer to daniel defoe. *Stockholm Studies in English*, 1974.

[155] H. Saevanee, N. Clarke, and S. Furnell. Sms linguistic profiling authentication on mobile device. In *Network and System Security (NSS), 2011 5th International Conference on*, pages 224–228. IEEE, 2011.

[156] T. Sammes and B. Jenkinson. The grppling of the sprms: Binary analysis of microsoft office documents. March 2012.

[157] Yukie Sano, Hideki Takayasu, and Misako Takayasu. Evaluation of latent vocabularies through zipfs law and heaps law. In *Proceedings of the European Conference on Complex Systems 2012*, pages 739–743. Springer, 2014.

[158] Minoru Sasaki and Hiroyuki Shinnou. Spam detection using text clustering. In *Cyberworlds, 2005. International Conference on*, pages 4–pp. IEEE, 2005.

[159] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. Practical network support for ip traceback. *ACM SIGCOMM Computer Communication Review*, 30(4):295–306, 2000.

[160] Stefan Savage, David Wetherall, Anna Karlin, and Tom Anderson. Practical network support for ip traceback. In *ACM SIGCOMM Computer Communication Review*, volume 30, pages 295–306. ACM, 2000.

[161] Jacques Savoy. Feature selections for authorship attribution. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, SAC '13, pages 939–941, New York, NY, USA, 2013. ACM.

[162] Elena Semino and Jonathan Culpeper. *Cognitive stylistics: Language and cognition in text analysis*, volume 1. John Benjamins Publishing, 2002.

[163] Bennett A Shaywitz, Sally E Shaywltz, Kenneth R Pugh, R Todd Constable, Pawel Skudlarski, Robert K Fulbright, Richard A Bronen, Jack M Fletcher, Donald P Shankweiler, Leonard Katz, et al. Sex differences in the functional organization of the brain for language. 1995.

[164] Stavros N Shiaeles and Maria Papadaki. Fhsd: An improved ip spoof detection method for web ddos attacks. *The Computer Journal*, 58(4):892–903, 2014.

[165] Douglas C Sicker and Lisa Blumensaadt. Misunderstanding the layered model (s). *J. on Telecomm. & High Tech. L.*, 4:299, 2005.

[166] J. Smith and I. Fujinaga. A review of authorship attribution. pages –, 2008.

[167] M Wilfrid A Smith. Hapax legomena in prescribed positions: an investigation of recent proposals to resolve problems of authorship. *Literary and Linguistic Computing*, 2(3):145–152, 1987.

[168] Smyth. The threat of social media. BBC, BBC Leicester, 2017.

[169] Smyth and Smith. Using open source intelligence techniques to highlight the dangers of an open social media profile. 2016.

[170] R. Smyth. A multidisciplinary approach to cyber attribution. In *Information Assurance Advisory Council Symposium* [170].

[171] A.C. Snoeren. Hash-based ip traceback. In *ACM SIGCOMM Computer Communication Review*, volume 31, pages 3–14. ACM, 2001.

[172] A.C. Snoeren, C. Partridge, L.A. Sanchez, C.E. Jones, F. Tchakountio, B. Schwartz, S.T. Kent, and W.T. Strayer. Single-packet ip traceback. *IEEE/ACM Transactions on Networking (ToN)*, 10(6):721–734, 2002.

[173] Lance Spitzner. *Honeypots: tracking hackers*, volume 1. Addison-Wesley Reading, 2003.

[174] Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, and Murat Demirbas. Short text classification in twitter to improve information filtering. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 841–842. ACM, 2010.

[175] E. Stamatatos. A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, 60(3):538–556, 2009.

[176] Ian Stewart and Vann Joines. *TA today: A new introduction to transactional analysis*. Lifespace Pub., 1987.

[177] Cliff Stoll. *The cuckoo's egg: tracking a spy through the maze of computer espionage*. SimonandSchuster. com, 2005.

[178] R. Stone. Centertrack: An ip overlay network for tracking dos floods. In *Proceedings of the USENIX Security Symposium*, pages 199–212, 2000.

[179] Michael Stubbs. Whorf's children: Critical comments on critical discourse analysis (cda). *British Studies in Applied Linguistics*, 12:100–116, 1997.

[180] Yutaka Sugawara, Mary Inaba, and Kei Hiraki. Over 10gbps string matching mechanism for multi-stream packet scanning systems. *Field Programmable Logic and Application*, pages 484–493, 2004.

[181] D Roger Tallentire. Towards an archive of lexical norms: A proposal. *The computer and literary studies*, pages 39–60, 1973.

[182] Alfred Edward Taylor. *Plato: The man and his work*. Courier Dover Publications, 2001.

[183] Ronald Thisted and Bradley Efron. Did shakespeare write a newly-discovered poem? *Biometrika*, 74(3):445–455, 1987.

[184] Fiona J Tweedie and R Harald Baayen. How variable may a constant be? measures of lexical richness in perspective. *Computers and the Humanities*, 32(5):323–352, 1998.

[185] Blase Ur, Pedro Giovanni Leon, Lorrie Faith Cranor, Richard Shay, and Yang Wang. Smart, useful, scary, creepy: perceptions of online behavioral advertising. In *proceedings of the eighth symposium on usable privacy and security*, page 4. ACM, 2012.

[186] Peter Van Kranenburg and Eric Backer. Musical style recognition-a quantitative approach. In *Proceedings of the Conference on Interdisciplinary Musicology (CIM)*, pages 106–107, 2004.

[187] Ron G Van Schyndel, Andrew Z Tirkel, and Charles F Osborne. A digital watermark. In *Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference*, volume 2, pages 86–90. IEEE, 1994.

[188] Robert D Van Valin. *Syntax: Structure, meaning, and function*. Cambridge University Press, 1997.

[189] Emmanouil Vasilomanolakis, Shreyas Srinivasa, and Max Mühlhäuser. Did you really hack a nuclear power plant? an industrial control mobile honeypot. In *Communications and Network Security (CNS), 2015 IEEE Conference on*, pages 729–730. IEEE, 2015.

[190] S Venkatramulu and CG Rao. Various solutions for address resolution protocol spoofing attacks. *International Journal of Scientific and Research Publications*, 3(7):1, 2013.

[191] Bin Wang and Wen-feng PAN. A survey of content-based anti-spam email filtering [j]. *Journal of Chinese Information Processing*, 5:000, 2005.

[192] Xinyuan Wang, Shiping Chen, and Sushil Jajodia. Network flow watermarking attack on low-latency anonymous communication systems. In *Security and Privacy, 2007. SP'07. IEEE Symposium on*, pages 116–130. IEEE, 2007.

[193] Xinyuan Wang, Douglas S Reeves, and S Felix Wu. Tracing based active intrusion response. *Journal of Information Warfare*, 1(1):50–61, 2001.

[194] R. Wei. A framework of distributed agent-based network forensics system. *Proceedings of DFRWS2004*, 99:100–, 2004.

[195] FL Wellman. The art of cross-examination (rev. and enl.). new york: Touchstone. adapted from: Dye. Technical report, TR (1995). Understanding public policy. Englewood Cliffs, NJ: Prentice-Hall, 1936.

[196] D.A. Wheeler. Techniques for cyber attack attribution. Technical report, 2003.

[197] Michael Whitman and Herbert Mattord. *Management of information security*. Nelson Education, 2013.

[198] Benjamin Lee Whorf. *Language, thought, and reality: Selected writings of Benjamin Lee Whorf*. Mit Press, 2012.

[199] Carrington B Williams. Mendenhall's studies of word-length distribution in the works of shakespeare and bacon. *Biometrika*, 62(1):207–212, 1975.

[200] Gordon S Wood. The authorship of the letters from the federal farmer. *The William and Mary Quarterly: A Magazine of Early American History*, pages 299–308, 1974.

[201] D. YAN, Y. WANG, SEN SU, and F. YANG. A precise and practical ip traceback technique based on packet marking and logging. pages –, 2016.

[202] Yun Yang, Hongli Yang, and Jia Mi. Design of distributed honeypot system based on intrusion tracking. In *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, pages 196–198. IEEE, 2011.

[203] Kunikazu Yoda and Hiroaki Etoh. Finding a connection chain for tracing intruders. In *ESORICS*, volume 1895, pages 191–205. Springer, 2000.

[204] George Udny Yule. *The statistical study of literary vocabulary*. CUP Archive, 1944.

[205] Michal Zalewski. p0f: Passive os fingerprinting tool, 2006.

[206] Qinghua Zhang, Douglas S. Reeves, Peng Ning, and S. Purushothaman Iyer. Analyzing network traffic to detect self-decrypting exploit code. In *Proceedings of the 2Nd ACM Symposium on Information, Computer and Communications Security*, ASIACCS '07, pages 4–12, New York, NY, USA, 2007. ACM.

[207] R. Zheng, J. Li, H. Chen, and Z. Huang. A framework for authorship identification of online messages: Writing style features and classification techniques. *Journal of the American Society for Information Science and Technology*, 57(3):378–393, 2006.

[208] XU Guo zhu. A new perspective of language study: Critical linguistics. , 9(011):679–685, 2011.

[209] Sebastian Zimmeck, Jie S Li, Hyungtae Kim, Steven M Bellovin, and Tony Jebara. A privacy analysis of cross-device tracking. In *26th USENIX Security Symposium, USENIX Security 17)*, pages 1391–1408. USENIX Association, 2017.

[210] George Kingsley Zipf. Selected studies of the principle of relative frequency in language. 1932.