2016

Evidential Reasoning for Forensic Readiness

Yi-Ching Liao
Norwegian University of Science and Technology

Hanno Langweg
Norwegian University of Science and Technology

Follow this and additional works at: http://commons.erau.edu/jdfsl
Part of the Computer Law Commons, and the Information Security Commons

Recommended Citation
DOI: https://doi.org/10.15394/jdfsl.2016.1372
Available at: http://commons.erau.edu/jdfsl/vol11/iss1/2
EVIDENTIAL REASONING FOR FORENSIC READINESS

Yi-Ching Liao and Hanno Langweg
Norwegian University of Science and Technology
Teknologivn. 22, 2815 Gjøvik, Norway
{yi-ching.liao2, hanno.langweg}@ntnu.no

ABSTRACT
To learn from the past, we analyse 1,088 “computer as a target” judgments for evidential reasoning by extracting four case elements: decision, intent, fact, and evidence. Analysing the decision element is essential for studying the scale of sentence severity for cross-jurisdictional comparisons. Examining the intent element can facilitate future risk assessment. Analysing the fact element can enhance an organization’s capability of analysing criminal activities for future offender profiling. Examining the evidence used against a defendant from previous judgments can facilitate the preparation of evidence for upcoming legal disclosure. Following the concepts of argumentation diagrams, we develop an automatic judgment summarizing system to enhance the accessibility of judgments and avoid repeating past mistakes. Inspired by the feasibility of extracting legal knowledge for argument construction and employing grounds of inadmissibility for probability assessment, we conduct evidential reasoning of kernel traces for forensic readiness. We integrate the narrative methods from attack graphs/languages for preventing confirmation bias, the argumentative methods from argumentation diagrams for constructing legal arguments, and the probabilistic methods from Bayesian networks for comparing hypotheses.

Keywords: legal knowledge extraction, forensic readiness, evidential reasoning, kernel tracing, attack description languages, argumentation diagrams, Bayesian networks

1. INTRODUCTION
“A case is only as strong as its evidence” (Torpey, 2009). Many computer crime cases are closed due to the lack of evidence, and the most common reason is not preparing for upcoming legal disclosure at all, which is equivalent to covering the tracks for perpetrators (Rowlingson, 2004). Moreover, covering the tracks has become the standard operating procedure for perpetrators (Graves, 2007). To meet the burden of proof, we should establish an information retention strategy to prepare for upcoming legal disclosure (International Organization for Standardization, 2015), instead of merely relying on the evidence remained.

Since digital evidence is more susceptible to tampering and subsequent modification than traditional documents, it causes more uncertainties in legal cases, such as using the Trojan horse defense to avoid a conviction (Brenner, Carrier, & Henninger, 2004). What is worse, forged digital evidence can result in miscarriages of justice, such as
framing victims by planting digital evidence. To prevent miscarriages of justice caused by false evidence, we have to consider evidence reliability early in the collection phase by choosing more reliable evidence sources, instead of merely securing evidence with tamper resistance and detection techniques later in the preservation phase.

To facilitate the preparation of reliable evidence before the occurrence of legal actions, we emphasise the domain of computer crime and analyse the “computer as a target” judgments for evidential reasoning through case element extraction in Section 2. Inspired by the effectiveness of case elements and the feasibility of transforming them into argumentation diagrams, we integrate the narrative methods, the argumentative methods, and the probabilistic methods to produce scenarios through attack graphs/languages, generate hypotheses through argumentation diagrams, and assess probabilities through Bayesian networks in Section 3. Section 4 concludes the paper with the discussion of the effectiveness of method integration for evidential reasoning.

2. ANALYSING JUDGMENTS FOR EVIDENTIAL REASONING

To learn from the past, we analyse 1,088 “computer as a target” judgments through language engineering techniques for evidential reasoning. We explain our analysis methods and findings as follows:

2.1 Related Work

Distinguished from previous studies on legal knowledge extraction, we emphasise the analysis of “computer as a target” judgments written in Chinese characters from four case elements: decision, intent, fact, and evidence. Table 1 presents a comparative analysis of previous research on legal text summarization and analysis from three perspectives: language, corpus, and elements. Previous studies mostly collect and analyse judgments written in English, and mainly focus on general criminal cases. To extend the research scope, we emphasise the domain of computer crime and analyse the “computer as a target” judgments for evidential reasoning.

2.2 Collecting “Computer as a Target” Judgments

The criminal activities targeting at computers demand more technical knowledge than the activities utilizing computers as tools. Therefore, we emphasise the analysis of “computer as a target” judgments for evidential reasoning. Since different jurisdictions have their own computer crime laws, we need to employ different search terms based on the corresponding computer crime laws for judgment collection.

Chinese judgments are available at the website wenshu.court.gov.cn (China Judgments Online) for public access. The Criminal Law of the People’s Republic of China clearly differentiates between the “computer as a target” crime and the “computer as a tool” crime in separate articles. Therefore, we employ “computer information system” as the search term for “case name” to collect judgments regarding “illegal invasion of computer information system” and “deleting, altering, adding or jamming the functions of the computer information system” (Council of Europe, 2008). We employ Scrapy (Scrapinghub, Ltd., 2015), an open source Web crawler, and utilize the XML path language to locate and retrieve the judgment content. We categorize the collected Chinese judgments into two judgment types: first instance and appeal. We
Table 1. A Summary of Research on Legal Text Summarization and Analysis

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Language</th>
<th>Corpus</th>
<th>Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gelbart and Smith</td>
<td>English</td>
<td>more than 1000 economic loss cases and general British Columbia cases</td>
<td>legal concepts, statute citations, case citations, and facts</td>
</tr>
<tr>
<td>Schweighofer et al.</td>
<td>English</td>
<td>75 full text documents of court decisions from the European Community law database</td>
<td>index with the thesaurus entries</td>
</tr>
<tr>
<td>Uyttendaele et al.</td>
<td>Dutch</td>
<td>more than 3000 decisions from the correctional Court of Leuven</td>
<td>superscription, victim, accused, alleged offenses, transition formulation, opinion of the court, legal foundations, verdict, and conclusion</td>
</tr>
<tr>
<td>Farzindar and Lapalme</td>
<td>English</td>
<td>3500 judgments from the Federal Court of Canada</td>
<td>four thematic segmentations: introduction, context, juridical analysis, and conclusion</td>
</tr>
<tr>
<td>Hachey and Grover</td>
<td>English</td>
<td>188 judgments from the UK House of Lords</td>
<td>seven rhetorical annotations: fact, proceedings, background, framing, disposal, textual, and other</td>
</tr>
<tr>
<td>Saravanan et al.</td>
<td>English</td>
<td>200 judgments related to rent control, income tax and sales tax</td>
<td>seven rhetorical roles: identifying the case, establishing facts of the case, arguing the case, history of the case, arguments, ratio of the decision, and final decision</td>
</tr>
<tr>
<td>Chieze et al.</td>
<td>EnglishFrench</td>
<td>14,380 historical decisions from Canadian federal courts and provincial tribunals</td>
<td>four thematic segments (Mailhot &amp; Carnwath, 1998): introduction, context, reasoning, and conclusion</td>
</tr>
<tr>
<td>Wyner</td>
<td>English</td>
<td>47 criminal cases drawn from the California Supreme Court and State Court of Appeal</td>
<td>case citations, names of parties, roles of parties, and final decision</td>
</tr>
<tr>
<td>Youssi-Monod et al.</td>
<td>EnglishFrench</td>
<td>3715 decisions from the Canadian courts</td>
<td>four decision sections: introduction, context, reasoning and conclusion</td>
</tr>
<tr>
<td>Galgani et al.</td>
<td>English</td>
<td>5,705 case reports from the Federal Court of Australia</td>
<td>collected catchphrases, such as courts and judges, corporations, costs, etc.</td>
</tr>
</tbody>
</table>

found 26 duplicated judgments, two judgments without content, and four irrelevant judgments while collecting Chinese judgments.

As for Taiwan, the Judicial Yuan makes legislative and judicial texts accessible at the website jirs.judicial.gov.tw, and we utilize “offenses against the computer security” as the search term for “cause of action” and “criminal case” as “judgment type” for judgment collection. Since certain character in the case number represents different judgment types, we can categorize the collected Taiwanese judgments into six judgment types: prosecution, summary judgment, appeal, civil action, private prosecution, and mediation. The most common types of Taiwanese judgments are prosecution and summary judgment, which form 88.9% of the collected Taiwanese judgments. The average character count of summary judgments is around half of the prosecution judgments, which are 2820.58, and 5801.13 respectively. We also found two duplicated judgments while collecting Taiwanese judgments. Until May 29th 2016, we totally collect 1,088 “computer as a target” judgments: 358 Chinese judgments and 730 Taiwanese judgments.

Figure 1 shows the number of “computer as a target” judgments by judgment year. The Criminal Law of the People’s Republic of China incriminates “illegal invasion of computer information system” and “deleting, altering, adding or jamming the functions of the computer information system” in 1997, whereas Criminal Code of the Republic of China incriminates “offenses against the computer security” in 2003. However, there is only one Chinese judgment concerning “computer as a target” crime before 2011, and the number of judgments collected in China is less than in Taiwan. The number of “computer as a target” judgments is much less than expected considering the number of incidents reported.

2.3 Extracting Case Elements

To analyse the “computer as a target” judgments for evidential reasoning, we determine four case elements for legal knowledge extraction: decision, intent, fact, and evi-
Examine the decision element is essential for studying the scale of sentence severity, which can serve as a criterion for cross-jurisdictional comparisons. Analysing the intent element is indispensable to threat analysis, which can facilitate future risk assessment. Examining the fact element can enhance an organization’s capability of understanding criminal behaviours for future offender profiling. Analysing the evidence used against a defendant from previous judgments can facilitate the preparation of evidence before the occurrence of legal actions.

To extract case elements, we employ five processing resources (Cunningham et al., 2014) provided by GATE (Cunningham, 2002), an open source language engineering software, to locate annotations from the “computer as a target” judgments, which we list as follows:

1. **RegEx Sentence Splitter** annotates sentences according to the files containing regular expressions of sentence splits, which we utilize as the indicators of different judgment stages: decision, fact, and reasoning. Regarding a judgment as a transaction, Cheng (2010) determines generic structures of Chinese and Taiwanese judgments by identifying various stages and legal functions. The orders of stages are different between Chinese and Taiwanese judgments, which are shown in Table 2. Chinese judgments establish the fact first, then justify the decision, and finally declare the decision. On the other hand, Taiwanese judgments declare the decision first, then establish the fact, and eventually justify the decision.

Despite different stage orders, the case elements exist within the same stages. For Chinese judgments, we utilize RegEx Sentence Splitter to locate the linguistic markers of the fact and evidence elements, which we later extract the first paragraph between them as the fact element by JAPE Transducer. As for Taiwanese judgments, we employ RegEx Sentence Splitter to locate the linguistic markers as the separator between the decision and intent elements.

2. **GATE Unicode Tokeniser** divides judgments into smaller tokens, such as punctuations and numbers. We customize GATE Unicode Tokeniser to annotate one or more occurrences of Unicode character between punctuations as a token, which we regard as a sentence for further higher-level annotations by JAPE Transducer.

3. **ANNIE Gazetteer** aims to search the pre-defined words for further annotation by JAPE Transducer. We set the “wholeWordsOnly” parameter to false to allow ANNIE Gazetteer to match not only whole words. Due to the different judgment structures, we employ different linguistic markers to different jurisdictions, which are summarized in Table 2. Currently we generate these linguistic markers manually through analysing the indicators of different judgment stages and case elements. We only update the linguistic markers to processing resources after
Table 2. The Analysis of Judgments for Case Element Extraction

<table>
<thead>
<tr>
<th>Stage</th>
<th>Element</th>
<th>Linguistic Markers</th>
<th>Processing Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>China</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fact</td>
<td>intent</td>
<td>intent → [purpose, for, need]</td>
<td>ANNIE Gazetteer</td>
</tr>
<tr>
<td>fact</td>
<td></td>
<td>the sentence which contains intent</td>
<td>JAPE Transducer</td>
</tr>
<tr>
<td>fact</td>
<td></td>
<td>fact, start → [the prosecutor accused, procuratorate accused, after being examined]</td>
<td>RegEx Sentence Splitter</td>
</tr>
<tr>
<td>reasoning</td>
<td>evidence</td>
<td>evidence, start → [the above fact, given with evidence, to support the above allegation]</td>
<td>JAPE Transducer</td>
</tr>
<tr>
<td>decision</td>
<td>decision</td>
<td>decision, start → [the decision is as follows, based on the defendant]</td>
<td>ANNI Gazetteer</td>
</tr>
<tr>
<td>decision</td>
<td>decision</td>
<td>after decision, start, the sentence which contains decision</td>
<td>JAPE Transducer</td>
</tr>
<tr>
<td>Taiwan</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fact</td>
<td></td>
<td>fact, start → [reason, crime fact]</td>
<td>RegEx Sentence Splitter</td>
</tr>
<tr>
<td>fact</td>
<td></td>
<td>guilt → [detention, imprisonment, probation, fine]</td>
<td>ANNIE Gazetteer</td>
</tr>
<tr>
<td>fact</td>
<td></td>
<td>length → [day, month, year, dollar]</td>
<td>JAPE Transducer</td>
</tr>
<tr>
<td>fact</td>
<td></td>
<td>other → [not guilty, dismissal, overrule, dismissal, jurisdictional error, summary judgment, grounds for appeal]</td>
<td>ANNI Gazetteer</td>
</tr>
<tr>
<td>reasoning</td>
<td>evidence</td>
<td>evidence, start → [based on, the above fact, the above crime fact, mainly based on, basis of crime fact]</td>
<td>JAPE Transducer</td>
</tr>
<tr>
<td>reasoning</td>
<td>evidence</td>
<td>evidence, ref → [judgment, as reference]</td>
<td>ANNI Gazetteer</td>
</tr>
<tr>
<td>reasoning</td>
<td>evidence</td>
<td>after evidence, start, the sentences which contains parenthenses</td>
<td>JAPE Transducer</td>
</tr>
<tr>
<td>reasoning</td>
<td>evidence</td>
<td>the paragraph which contains evidence, start</td>
<td>ANNI Gazetteer</td>
</tr>
<tr>
<td>reasoning</td>
<td>evidence</td>
<td>the sentence which contains evidence, ref</td>
<td>JAPE Transducer</td>
</tr>
</tbody>
</table>

verifying the enhanced annotation coverage. However, the diverse usages of legal terms by different judges obstruct the automatic generation of linguistic markers.

4. **JAPE Transducer** recognises higher-level annotations through regular expressions. For example, after we utilize **ANNIE Gazetteer** to locate the linguistic markers for the start of the decision element “the decision is as follows” and the keyword of the decision element “is sentenced” from Chinese judgments, we extract the sentences which contain “is sentenced” after “the decision is as follows” as the decision elements by **JAPE Transducer**.

5. **Flexible Exporter** saves the annotations to files. After locating four case elements from judgments, we employ **Flexible Exporter** to store the annotations for further analysis and visualization.

### 2.4 Verifying Annotation Coverage

Table 3 presents the number of judgments collected, the average character count, and the number of case element not found according to different judgment types. Since there is no frequently used term to indicate the intent in Chinese judgments, we fail to extract the intent element from 134 Chinese judgments, from which we can observe that the pre-defined words in **ANNIE Gazetteer** lists greatly influence the accuracy of element extraction. Analysing the annotation coverage from the judgment types, we fail to extract almost all the case elements from 26 Taiwanese judgments of civil action and mediation types due to the lack of element in these judgments. Moreover, since we utilize **RegEx Sentence Splitter** to locate the linguistic markers as the separator between the decision and intent elements for Taiwanese judgments, extracting the decision element will fail if there is no separator for **RegEx Sentence Splitter** to
locate. Not every Taiwanese judgment has the same judgment stages, even though Taiwanese judgments have more identifiable linguistic markers to separate judgment stages than Chinese judgments.

Table 3. A Summary of Collected Chinese and Taiwanese Judgments

<table>
<thead>
<tr>
<th>Judgment Type</th>
<th>China Avg. Char Count</th>
<th>Taiwan Avg. Char Count</th>
<th>Number of Case Elements not Found</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Instance Decision</td>
<td>4160.06</td>
<td>5922.81</td>
<td>124 72 22</td>
</tr>
<tr>
<td>Appeal</td>
<td>5922.81</td>
<td>66 32 24 12 17</td>
<td></td>
</tr>
<tr>
<td>Prosecution</td>
<td>5801.13</td>
<td>331 32 115 121</td>
<td></td>
</tr>
<tr>
<td>Summary Judgment</td>
<td>2820.58</td>
<td>318 14 97 97</td>
<td></td>
</tr>
<tr>
<td>Appeal</td>
<td>5138.15</td>
<td>58 2 13 13 22</td>
<td></td>
</tr>
<tr>
<td>Civil Action</td>
<td>677.52</td>
<td>23 15 23 25 25</td>
<td></td>
</tr>
<tr>
<td>Private Prosecution</td>
<td>314.35</td>
<td>71 1 1 1 2</td>
<td></td>
</tr>
<tr>
<td>Mediation</td>
<td>513.00</td>
<td>3 1 1 1 3 9</td>
<td></td>
</tr>
</tbody>
</table>

2.5 Judgment Analysis Findings

After analysing 1,088 “computer as a target” judgments through case element extraction, we describe the major findings as follows:

2.5.1 Effectiveness of Case Elements

The decision element serves as a good criterion to analyse the scale of sentence severity for cross-jurisdictional comparisons. Regarding not guilty as zero days, the average length of imprisonment is 962.07 days based on the decision elements extracted from Chinese judgments. On the other hand, the average length of imprisonment is 170.46 days according to Taiwanese decision elements, which is not that severe compared to the maximum three or five years imprisonment regulated in criminal code on offenses against the computer security. We can observe the defendants who commit computer crime tend to be sentenced more severely in China than those in Taiwan, even though the number of judgments collected in China is less than in Taiwan. Note that the written numbers are different between Taiwanese and Chinese judgments; most of Taiwanese judgments use financial characters, whereas Chinese judgments use normal characters.

Studying the intent element is beneficial for future risk assessment, even though it is difficult to have solid evidence to prove the intent of a defendant. Since threat is one of the indispensable components of risk, and threat possesses intent and capability, examining the intent element is advantageous for risk assessment and determining the level of security control to protect information assets. According to 224 intent elements extracted from Chinese judgments, the most common intent is to gain the illegal benefits. On the other hand, based on 474 intent elements extracted from Taiwanese judgments, the intent “for purpose to exercise unlawful control over others property” is more popular than the intent of gaining the illegal benefits.

The fact elements extracted are well constructed and describe the events that previously occurred in narrative forms, which can facilitate the analysis of criminal activities for further crime scenario construction and offender profiling. Based on 343 fact elements extracted from Chinese judgments, the common activities include installing Trojan horse for stealing money from on-line accounts, and even selling bots for distributed denial-of-service attacks. According to 472 fact elements extracted Taiwanese judgments, the most common behaviour is the illegal possession of others’ accounts.

Analysing the evidence element facilitates the preparation of evidence before the occurrence of legal actions. An organization should establish an information retention strategy to prepare for upcoming legal disclosure (International Organization for Standardization, 2015), which can be easily clarified by analysing the evidence used against a defendant from previous judgments. However, based on the total 651 evidence elements extracted Chinese and Taiwanese judgments, we observe the digital evidence...
supporting the “computer as a target” cases rarely stands alone. Even the “computer as a target” cases still depend heavily on non-digital evidence, such as testimonials, documents, or confessions.

### 2.5.2 Transformation into Argumentation Diagrams

Based on the case elements extracted from judgments, we can follow the concepts of datum and claim from the Toulmin model of argument (Toulmin, 2003), which are shown in Figure 2, to develop an automatic judgment summarizing system. Taking the file generated by Flexible Exporter as input, we visualize the case elements with DOT, a plain text graph description language, and employ Graphviz (Ellson, Gansner, Koutsofios, North, & Woodhull, 2001), an open source graph drawing tool, to convert the DOT files to PDF files.

![Figure 2. Transformation of Case Elements into Argumentation Diagrams](image)

Figure 3 presents an overview of the measures and procedures for automatic judgment summarization. Figure 4 presents an automatically generated argumentation diagram from a Chinese judgment. Using rectangles to represent a fact or an observation and ovals to present a consequent assertion (Shum, 2003), we summarize the “computer as a target” judgments automatically in a consistent format, which can enhance the accessibility and readability of judgments. Since the fact element describes the previously occurred events in chronological order, we divide the fact element into shorter phrases, and list the phrases from top to bottom for better comprehensibility of criminal activities. We also categorize the evidence element into digital evidence and non-digital evidence to demonstrate that the “computer as a target” judgments do not depend heavily on digital evidence as expected.

![Figure 3. An Overview of an Automatic Judgment Summarizing System](image)

### 2.5.3 Grounds of Inadmissibility

The gathered evidence only become admissible when it meets the criteria of relevance and reliability. Based on collected Chinese and Taiwanese judgments, the grounds of inadmissibility include processed by unqualified personnel, diverse malware analysis result, inaccurate calculation of monetary value, no evidential connection, reasonable doubt, etc. Analysis outcomes of the admissibility and inadmissibility of evidence from
the “computer as a target” judgments are valuable inputs for probability assessment. Extracting 5W1H information and other factors affecting the scale of sentence severity can also assist the calculation of probabilities.

2.5.4 Need for Forensic Readiness

Since the criminal activities targeting at computers demand more technical knowledge than general criminal cases, the “computer as a target” judgments are supposed to depend heavily on digital evidence. However, based on the evidence elements extracted, we discover the digital evidence rarely stands alone, even for supporting the “computer as a target” cases. Moreover, since the number of incidents reported are much more than the number of judgments, there must be many “computer as a target” cases closed before being brought to courts. It is essential for organisations to prepare themselves for digital forensics with reliable digital evidence, in other words- forensic readiness.

3. EVIDENTIAL REASONING OF KERNEL TRACES

To learn from the past, we analyse previous “computer as a target” judgments for evidential reasoning in Section 2. To set out the future, organisations can prepare themselves for upcoming legal disclosure with reliable digital evidence, such as kernel traces, the low-level execution logs of operating systems (Giraldeau, Desfossez, Goulet, Dagenais, & Desnoyers, 2011). To conduct evidential reasoning of kernel traces for forensic readiness, we propose a framework by integrating the narrative methods from attack graphs/languages, the argumentative methods from argumentation diagrams, and the probabilistic methods from Bayesian networks, which is shown in Figure 5.

3.1 Related Work

Distinguished from previous studies on method integration for evidential reasoning and interpretation, which are summarized in Table 4, we employ the narrative methods from attack graphs and attack description languages to emphasise the investigation of information security incidents in this paper.

Table 4. A Summary of Method Integration for Evidential Reasoning

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Narrative</th>
<th>Argumentative</th>
<th>Probabilistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hepler et al.</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>F. J. Bex et al.</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Condliffe et al.</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Vlek et al.</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Verheij</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Timmer et al.</td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>
3.2 Producing Scenarios through Attack Graphs/Languages

Producing scenarios automatically can reduce the possibility of bias and errors. Once a suspect has been targeted, the evidence gathering process tends to become restricted to supporting the guilt of the suspect. To keep investigators from the confirmation bias, we suggest to employ the essential elements of attack description languages (Cuppens & Ortalo, 2000; Templeton & Levitt, 2000; Cheung, Lindqvist, & Fong, 2003): pre-conditions (require capabilities) and post-conditions (provide capabilities) as datums, and scenarios as claims, to produce scenarios automatically from kernel traces.

Attack graphs aim to represent the potential intrusion paths for security vulnerability analysis, whereas attack description languages usually address the generic description issues for knowledge interchange. Despite the differences, both of them are effective narrative methods to demonstrate information security incidents. Table 5 summarizes the elements to present information security incidents in a narrative form. The pre-condition is the set of resources required for an attack to occur (e.g., the host vulnerability information and the communication connectivity), the scenario is the set of events of an attack, and the post-condition is the set of resources obtained by a successful attack.

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Graph/Language Elements</th>
<th>Nodes</th>
<th>Edges</th>
<th>Edge Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phillips and Swiler</td>
<td>stages of attack, actions, conditions</td>
<td>privileges, success probability</td>
<td>actions, conditions</td>
<td>actions, conditions</td>
</tr>
<tr>
<td>Ortalo et al.</td>
<td>privileges</td>
<td>vulnerabilities, efforts required</td>
<td>privileges, vulnerabilities, capabilities</td>
<td>actions, conditions</td>
</tr>
<tr>
<td>Cuppens and Ortalo</td>
<td>pre-condition, post-condition, scenario, detection, verification</td>
<td>pre-condition, post-condition, scenario, detection, verification</td>
<td>pre-condition, post-condition, scenario, detection, verification</td>
<td>pre-condition, post-condition, scenario, detection, verification</td>
</tr>
<tr>
<td>Ritchey and Ammann</td>
<td>hosts, connectivity matrix, exploits</td>
<td>hosts, connectivity matrix, exploits</td>
<td>hosts, connectivity matrix, exploits</td>
<td>hosts, connectivity matrix, exploits</td>
</tr>
<tr>
<td>Templeton and Levitt</td>
<td>require capabilities, provide capabilities, actions</td>
<td>require capabilities, provide capabilities, actions</td>
<td>require capabilities, provide capabilities, actions</td>
<td>require capabilities, provide capabilities, actions</td>
</tr>
<tr>
<td>Eckmann et al.</td>
<td>states, transitions</td>
<td>states, transitions</td>
<td>states, transitions</td>
<td>states, transitions</td>
</tr>
<tr>
<td>Michel and M</td>
<td>exploit, detection, responses</td>
<td>exploit, detection, responses</td>
<td>exploit, detection, responses</td>
<td>exploit, detection, responses</td>
</tr>
<tr>
<td>Cheung et al.</td>
<td>pre-condition, post-condition</td>
<td>pre-condition, post-condition</td>
<td>pre-condition, post-condition</td>
<td>pre-condition, post-condition</td>
</tr>
<tr>
<td>Noel et al.</td>
<td>exploits, conditions</td>
<td>dependencies, hardening costs</td>
<td>dependencies, hardening costs</td>
<td>dependencies, hardening costs</td>
</tr>
<tr>
<td>Dris et al.</td>
<td>attack goal, detection, fix</td>
<td>dependencies, fix</td>
<td>dependencies, fix</td>
<td>dependencies, fix</td>
</tr>
</tbody>
</table>

The scenario producing process requires the knowledge similar to the database of intrusion detection systems, which holds the security-related events correlated based on different system resources (e.g., file system, process management, and network). Using system resources for correlation not only increases the possibility of integrating kernel traces from other operating systems, but also enhances the comprehensibility of attack stages. Since the post-condition of a scenario can be the pre-condition of the following scenario, it is effortless to understand the intent of acquiring specific system resources.

For instance, if the database of security-related events defines the scenario of gathering information as pre-conditions of initiating a connection on a socket and check-
ing user’s permissions for a file, and post-
conditions of sending a message on a socket
and changing ownership of a file, we can pro-
duce the scenario of gathering information
automatically from gathered kernel traces.
Similarly, we can produce the scenario of
covering tracks with pre-conditions of exe-
cuting program and changing permissions of
a file, and post-conditions of sending signal
to a process and deleting a directory.

3.3 Generating Hypotheses
through Argumentation
Diagrams

Convicting a defendant demands construct-
ing legal arguments. To assist investiga-
tors to construct legal arguments, we sug-
gest to employ the concepts of datum and
claim from the Toulmin model of argument
(Toulmin, 2003) and legal idioms, such as
evidence-accuracy idioms and intent idioms
(Fenton, Neil, & Lagnado, 2013), for hypoth-
esis generation.

Argumentation diagrams aim to represent
the structure of an argument visually, which
facilitate the crime scenario construc-
tion, crime investigation, and decision sup-
port from a legal perspective. However, ar-
gument diagramming requires facts, argu-
ments, and evidence as input data, which
mainly depends on manual user input. Ta-
ble 6 summarizes the elements to represent
information security incidents in an argu-
mentative form.

Table 6. A Summary of Argumentation Di-
agrams

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Graph Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goodwin</td>
<td>facts, processes</td>
</tr>
<tr>
<td>Toulmin</td>
<td>datum, warrant, claim, backing, rebuttal</td>
</tr>
<tr>
<td>Verheij</td>
<td>statements</td>
</tr>
<tr>
<td>Reed and Rowe</td>
<td>reconstructed enthymemes, refutations</td>
</tr>
<tr>
<td>Chesevar et al.</td>
<td>claims, applications</td>
</tr>
<tr>
<td>Keppens and Schafer</td>
<td>facts, evidence, assumptions, hypothesis</td>
</tr>
<tr>
<td>F. Bex et al</td>
<td>data, inference, scheme</td>
</tr>
<tr>
<td>Gordon</td>
<td>statements, testimonial evidence</td>
</tr>
</tbody>
</table>

An automatic judgment summarizing sys-
tem, which we have developed in Section 2,
can minimize the manual effort required for
hypothesis generation and provide knowl-
edge for developing legal idioms from scenar-
ios. Examining the evidence used against
a defendant from previous judgments con-
stantly can also solidify legal cases and avoid
repeating past mistakes. The decision ele-
ment corresponds to hypotheses, the intent
and fact elements fit into legal idioms, and
the scenarios can be considered as evidence
in narrative forms. The legal idioms devel-
oped can establish attack or support rela-
tions with the hypotheses generated, which
are normally mutually exclusive. For exam-
ple, Figure 5 consists of two mutually exclu-
sive hypotheses: the intruder is guilty or not
guilty.

3.4 Assessing Probabilities
through Bayesian
Networks

From a legal perspective, Bayesian networks
can facilitate the interpretation of compli-
cated relations between evidence, the mod-
elling of legal arguments and crime scenarios,
and the process of evidential reasoning. To
assist investigators to discover of the most
supported hypotheses, we suggest to employ
the qualitative probability (Keppens, 2007)
from Bayesian networks for probability as-
essment.

Bayesian networks aim to present vari-
ables and their probabilistic relations in a
directed acyclic graph. Table 7 summarizes
the elements to present information security
incidents in a probabilistic form. Even
though Bayesian networks can present vari-
ables and their probabilistic relations in accu-
rate numbers, these numerical probabili-
ties demand huge amount of knowledge as
input for calculation.

The gathered evidence only become ad-
missible when it meets the criteria of rele-
vance and reliability. Since the developed
Table 7. A Summary of Bayesian Networks Regarding Legal Reasoning

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Graph Elements</th>
<th>Nodes</th>
<th>Edges</th>
<th>Edge Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zukerman et al.</td>
<td>propositions, inferences</td>
<td>n/a</td>
<td></td>
<td>n/a</td>
</tr>
<tr>
<td>Hepler et al.</td>
<td>modules, variables, inferred hierarchy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keppens</td>
<td>variables, influences</td>
<td></td>
<td></td>
<td>qualitative derivatives</td>
</tr>
<tr>
<td>F. J. Bex et al.</td>
<td>observations, arguments, evidence, inference</td>
<td></td>
<td></td>
<td>n/a</td>
</tr>
<tr>
<td>Keppens</td>
<td>variables</td>
<td></td>
<td></td>
<td>qualitative derivatives</td>
</tr>
<tr>
<td>Fenton et al.</td>
<td>basic causal structures, inferred hierarchy</td>
<td></td>
<td></td>
<td>probability</td>
</tr>
<tr>
<td>Vlek et al.</td>
<td>scenarios</td>
<td></td>
<td></td>
<td>n/a</td>
</tr>
<tr>
<td>Verheij</td>
<td>hypotheses, evidence</td>
<td></td>
<td></td>
<td>strengths</td>
</tr>
<tr>
<td>Timmer et al.</td>
<td>support factors, support, likelihood ratio</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

legal idioms attack or support the generated hypotheses, the datums and the claims connected to the legal idioms should be considered relevant. As for reliability, since a datum is a fact or an observation, and a claim is a consequent assertion that depends on datums (Shum, 2003), based on the object oriented concept (Hepler et al., 2007), we suggest to assign a reliability module and a qualitative probability module to each datum and claim respectively.

A reliability module should contain a conditional probability table from Bayesian networks, which can be derived from potential errors identified and vulnerability assessment outcomes. For instance, we can use the detection rate and false alarm rate as input for the reliability module of the intrusion detection system database required for producing scenarios. Thus, we can make errors present and transparent for a fair trial.

Since the comparison between different hypotheses does not require precise numerical probabilities, the qualitative approach is enough in the context of forensic analysis (Keppens, 2007). We suggest employing the analysis of inadmissibility grounds as input for calculation, assigning a qualitative probability module to a claim, and using different levels of edge thickness to indicate different levels of probability. For instance, we can utilize sliders to present scales of probability, which are shown in Figure 5.

4. CONCLUSIONS AND FUTURE WORK

Inspired by the feasibility of extracting legal knowledge for argument construction and employing grounds of inadmissibility for probability assessment, we integrate the narrative methods from attack graphs/languages, the argumentative methods from argumentation diagrams, and the probabilistic methods from Bayesian networks, to conduct evidential reasoning of kernel traces for forensic readiness.

Employing the elements of pre-condition and post-condition from attack description languages, we can keep investigators from confirmation bias, clarify the stages of attacks, and prevent miscarriages of justice. Following the concepts of datum and claim from argumentation diagrams, we can assist investigators to construct legal arguments. Moreover, the automatic judgement summarizing system developed can help investigators to solidify legal cases and avoid repeating past mistakes. Utilizing the ideas of legal idioms and qualitative probability from Bayesian networks, we can assist investigators to discover of the most supported hypotheses and assess the admissibility of gathered evidence.

In addition to broadening the judgment analysis scope for conducting comprehensive cross-jurisdictional comparisons, we plan to conduct a case study for evaluating the effectiveness of the proposed framework, and broaden the element scope for deeper legal knowledge extraction, such as 5W1H information. We also need to ensure the privacy implications for users are minimized to acceptable levels to alleviate the conflicts between accountability and privacy. We will make the raw data and source code available to interested researchers under an appropriate open source licenses.

© 2016 ADFSL
ACKNOWLEDGEMENT

Yi-Ching Liao is supported by the COINS Research School of Computer and Information Security.

REFERENCES


Detection (pp. 197–216). Springer Berlin Heidelberg. (DOI: 10.1007/3-540-39945-3,13)


