

Formalising Event Reconstruction in Digital Investigations

Pavel Gladyshev

The thesis is submitted to University College Dublin for the
degree of PhD in the Faculty of Science

August 2004

Department of Computer Science

Head of department: Prof. Barry Smyth
Supervisor: Dr. Ahmed Patel

TO MY FAMILY

Contents

List of Figures	vii
Abstract	ix
Declaration	x
Acknowledgements	xi
1 Introduction	1
1.1 Motivation	1
1.2 Research objectives	3
1.3 Research idea	3
1.4 Dissertation structure	4
1.5 Summary of achievements	6
2 Legal view of digital evidence	8
2.1 Legal concepts	8
2.1.1 The nature of disputes resolved in courts	8
2.1.2 The nature of legal proof	9
2.1.3 Standards of proof	10
2.1.4 Presumptions of fact	10
2.1.5 Burden of proof	11
2.1.6 Characteristics of evidence	11
2.1.7 Classes of evidence	12
2.2 Forensic science and evidence	14
2.2.1 Requirements to scientific evidence	14
2.3 Digital evidence	15
2.3.1 Anonymity of digital information	16
2.3.2 Context of digital information	16
2.3.3 Automated interpretation of digital information	17
2.3.4 Danger of damaged information	17
2.4 Summary	18

3	Concepts of digital forensics	19
3.1	Investigative process	19
3.2	Examination and analysis techniques	21
3.2.1	Search techniques	21
3.2.2	Reconstruction of events	24
3.2.3	Time analysis	30
3.3	Summary	33
4	The need for a theory of event reconstruction	34
4.1	Why digital forensics need a theory of event reconstruction . . .	35
4.2	State of the art	35
4.2.1	Attack trees	36
4.2.2	Visual investigative analysis	38
4.2.3	Multilinear events sequencing	40
4.2.4	Why-because analysis	43
4.3	Summary and research problem statement	46
4.3.1	Analysis of the state of the art	46
4.3.2	Research problem statement	47
5	Theoretical background	49
5.1	Formal notation	49
5.1.1	Mathematical notation	49
5.1.2	ACL2 notation	52
5.2	State machine model of computation	59
5.2.1	Basic state machine model and its variations	59
5.2.2	Creation of system models	67
5.2.3	Analysis of finite computations	69
5.3	Summary	74
6	Formalisation of event reconstruction problem	75
6.1	Informal example of state machine analysis	75
6.1.1	Investigation at ACME Manufacturing	75
6.1.2	Informal analysis illustrated with a state machine	77
6.1.3	Evidential statements	79
6.1.4	Assumption about reliability of evidence	81
6.2	Formalisation of event reconstruction problem	82
6.2.1	Finite state machine	82
6.2.2	Run	83
6.2.3	Partitioned run	84
6.2.4	Formalisation of backtracing	84
6.2.5	Formalisation of evidence	85
6.3	Summary	92

7	Event reconstruction algorithm	94
7.1	Computing the meaning of a fixed-length observation sequence . . .	95
7.2	Computing the meaning of a generic observation sequence	97
7.3	Computing the meaning of an evidential statement	99
7.4	Running time of event reconstruction algorithm	102
7.4.1	Prefix based representation of computation sets	102
7.4.2	An upper bound on the running time of <i>SolveFOS</i>	105
7.4.3	An upper bound on the running time of <i>SolveOS</i>	108
7.4.4	An upper bound on the running time of <i>SolveES</i>	110
7.5	Implementation of the event reconstruction algorithm	115
7.6	Summary	116
8	Evaluation	118
8.1	Evaluation criteria	119
8.1.1	Effectiveness of event reconstruction	120
8.1.2	Efficiency of event reconstruction	120
8.1.3	Legal admissibility of event reconstruction	121
8.2	Comparison with other event reconstruction techniques	123
8.3	Examples of formalised and automated event reconstruction . . .	123
8.3.1	Example 1. Networked printer analysis	125
8.3.2	Example 2. Example of event time bounding	137
8.4	Summary	156
9	Conclusions and future work	158
9.1	Problem	158
9.2	Solution	160
9.3	Lessons of the project	161
9.3.1	Achievements	161
9.3.2	Problems encountered	162
9.4	Future work	164
9.4.1	Extending formalisation of event reconstruction	164
9.4.2	Developing more efficient event reconstruction algorithm	165
9.4.3	Investigating new ways of constructing system models	166
9.4.4	Developing practical applications	166
9.5	Summary	167
	Bibliography	170
A	Selected ACL2 functions and macros	179
A.1	Functions	179
A.1.1	Logical functions	179
A.1.2	Integer functions	180
A.1.3	Functions for manipulating ordered pairs	180
A.1.4	Functions for manipulating lists	181
A.2	Macros	181

B Prefix based representation of computation sets	182
B.1 Prefix based representation of computation sets	182
B.2 Basic properties of prefix lists	183
C Source code	187
C.1 fd.lisp	187
C.2 util.lisp	193
C.3 rec.lisp	194
C.4 acme.lisp	199
C.5 ft.lisp	203
C.6 slack.lisp	207
C.7 draw.lisp	210
D Evidence of publication	212

List of Figures

1.1	Event reconstruction by backtracing transitions	4
1.2	Dissertation structure	5
3.1	Stages of investigative process	21
3.2	An example of time bounding	32
4.1	Attack tree describing different ways to open a safe	37
4.2	Example VIA chart	40
4.3	Example MES-diagram	42
4.4	WB-graph for the tarts rhyme	44
5.1	Well founded order	57
5.2	Counting state machine	60
5.3	A 2-bit binary counter	61
5.4	Interleaving model of concurrent system	63
5.5	Turing machine	66
5.6	A naive algorithm for finite computation analysis	70
5.7	Data abstraction	74
6.1	ACME Manufacturing LAN topology	76
6.2	Transition graph of the print job directory model	78
6.3	Evidence in ACME investigation	80
6.4	Run of computation	84
6.5	Functions ψ , ψ^{-1} , and Ψ^{-1}	86
6.6	A run that gives two explanations to an observation sequence	89
6.7	Evidential statement and related notions	91
7.1	Finding explanations of a fixed-length observation sequence	96
7.2	Computing the meaning of a fixed-length observation sequence	97
7.3	Computing the meaning of a generic observation sequence	99
7.4	Computing the meaning of an evidential statement	101
7.5	Sample output of the program	116
8.1	Comparison with other event reconstruction techniques	124
8.2	ACME Manufacturing LAN topology	125
8.3	Transition graph of the print job directory model	128

LIST OF FIGURES

8.4 Meaning of os_{Alice} with $infinitem = 2$ 133

8.5 Meaning of restricted Alice’s claim os'_{Alice} with $infinitem = 3$. . . 134

8.6 Meaning of evidential statement es_{ACME} with $infinitem = 6$. . . 136

8.7 Formation of the slack space 139

8.8 Times of transitions 140

8.9 Finding observations that happened before given observation . . . 145

8.10 Calculating the earliest possible time of given observation 147

8.11 Finding observations that happened after given observation 148

8.12 Calculation of the latest possible time of given observation 149

8.13 State machine model of the last cluster in a file 149

8.14 One step of event reconstruction of observation sequence os_{final} . . . 152

B.1 Algorithm for computing $IntersectPrefixes(x, y)$ 185

B.2 Algorithm for computing $X \cap Y$ 186

B.3 Algorithm for computing $\Psi^{-1}(X)$ 186

Abstract

The highly technical nature of computer crime facilitated the development of a new branch of forensic science called digital forensics. Instead of dead bodies, it collects and analyses data produced, transmitted, and stored by digital devices. The field of digital forensics is rapidly evolving. A major research challenge perceived by the digital forensic community is the need for theoretical basis validating correctness of methods and tools used by digital forensic investigators.

An important part of digital forensic analysis is event reconstruction. It is the process of determining the events that happened during the incident. In digital forensic investigations, event reconstruction is fairly complex. A single push of a button triggers a chain of events inside one or more digital devices that produce the digital evidence. Informal, unaided reasoning is not always sufficient to comprehensively analyse this chain of events.

One way to make event reconstruction more objective and rigorous is to employ mathematics. As a first step in this direction, this research aimed to give formal meaning to the problem of event reconstruction. More specifically, the objectives of this research were (1) to define a formal model of event reconstruction, and (2) to demonstrate that that model can be used as a basis for formalisation and automation of selected examples of digital forensic analysis.

To achieve these objectives, the following approach was adopted. First, a study of digital forensic techniques and legal theory was undertaken to clarify the requirements to and place of event reconstruction in digital forensic analysis. Then, a review of existing event reconstruction techniques was carried out. The review has shown that none of these technique are fully adequate in digital forensic context. Next, a formal model of event reconstruction was defined. The defined model possesses the following features:

- The system under investigation is modeled as a finite state machine.
- A special-purpose formalism called “evidential statement” is used for describing the evidence.
- The outcome of event reconstruction is given precise mathematical meaning in terms of the finite state machine model of the system.

The usefulness of the proposed model was then demonstrated by developing a generic event reconstruction algorithm, based on the defined model, and using that algorithm to formalise and automate selected examples of digital forensic analysis. Finally, several possible directions for future research have been suggested.

Declaration

I declare that this thesis is my own work and has not been submitted in any form for another degree or diploma at this, or any other, University or institute of tertiary education.

Pavel Gladyshev

August, 4 2004