

ENHANCEMENT OF A HIGH INTERACTION POSTGRESQL HONEYPOT WITH LLM
ASSISTED BEHAVIORAL ANALYSIS

By
Jundi Al Hafidz
200001310

MASTER'S DEGREE
in
MASTER OF INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING AND INFORMATION TECHNOLOGY



SWISS GERMAN UNIVERSITY
The Prominence Tower
Jalan Jalur Sutera Barat No. 15, Alam Sutera
Tangerang, Banten 15143 - Indonesia

July 2026

ENHANCEMENT OF A HIGH INTERACTION POSTGRESQL HONEYPOT WITH LLM
ASSISTED BEHAVIORAL ANALYSIS

By
Jundi Al Hafidz
200001310

MASTER'S DEGREE
in
MASTER OF INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING AND INFORMATION TECHNOLOGY



SWISS GERMAN UNIVERSITY
The Prominence Tower
Jalan Jalur Sutera Barat No. 15, Alam Sutera
Tangerang, Banten 15143 - Indonesia

July 2026
Revision after the Thesis Defence on January 15th 2026

STATEMENT BY THE AUTHOR

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

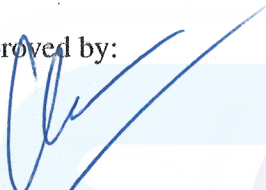


Jundi Al Hafidz

Student

15 August 2026

Approved by:



Dr. Ir. Charles Lim, B.Sc., M.Sc.

Thesis Advisor

15 August 2026



Dr. Ir. Eka Budiarto, S.T., M.Sc.

Thesis Co-Advisor

15 August 2026

Dr. Maulahikmah Galinium, S.Kom., M.Sc.

Dean of Faculty of Engineering and
Information Technology

15 August 2026

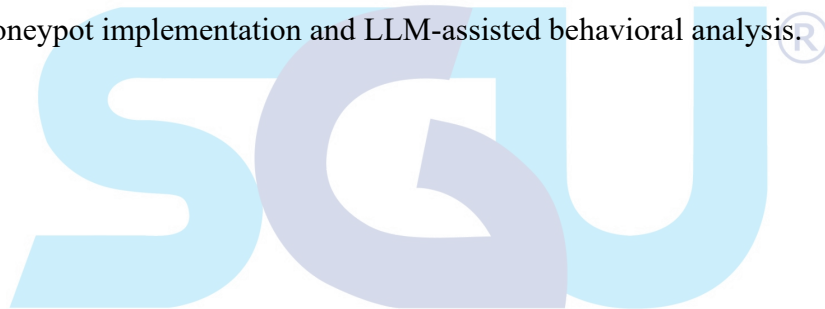
RELATED PUBLICATION

Publication Related to This Thesis

A related publication arising from the PostgreSQL honeypot research reported in this thesis has been published in IEEE Xplore:

J. Al Hafidz, C. Lim, and K. E. Silaen, “Enhancing Detection PostgreSQL Shell Honeypot with Interactive Query Response,” IEEE Xplore, Document 11384049.
Available: <https://ieeexplore.ieee.org/document/11384049/>

This thesis presents the complete master’s research, including its expanded high-interaction PostgreSQL honeypot implementation and LLM-assisted behavioral analysis.



ABSTRACT

ENHANCEMENT OF A HIGH INTERACTION POSTGRESQL HONEYPOT WITH LLM ASSISTED BEHAVIORAL ANALYSIS

By

Jundi Al Hafidz

Dr. Charles Lim, B.Sc., M.Sc., Advisor

Dr. Ir. Eka Budiarto, S.T., M.Sc., Co-Advisor

SWISS GERMAN UNIVERSITY



PostgreSQL has become an attractive target for attackers as its adoption grows across enterprise and cloud environments, yet common database defenses such as vulnerability scanning, log review, and penetration testing are largely reactive and reveal little about how intruders actually behave once they reach a database. This thesis enhances a high-interaction PostgreSQL honeypot and pairs it with LLM-assisted behavioral analysis so that raw attacker interaction can be turned into structured, evidence-based threat intelligence. The work is guided by the hypothesis that raising the honeypot's interaction level captures deeper, session-level attacker evidence that a shallow honeypot cannot, and that a hybrid of a deterministic rule layer and a large language model, combined through soft voting, detects and interprets attacks more reliably than either component used alone because the two make errors on different sessions. The system was built in three phases: a medium-interaction sensor that establishes baseline telemetry, a high-interaction deployment backed by an isolated Dockerized PostgreSQL that sustains realistic attacker sessions, and an analysis layer that classifies attacker behavior and maps it along a MITRE ATT&CK, CAPEC, and CVE/CWE evidence chain. During live deployment the

honeypot captured 827 query events, which were reconstructed into 161 real captured sessions (139 severe and 22 non-severe reconnaissance or protocol-noise sessions) and scored against a fixed, manually labelled reference. The deterministic rule layer achieved F1 0.985 without a false positive, while the LLM detected every severe session but over-flagged some negative-class sessions. Their soft-voting hybrid achieved F1 1.000 on the tabulated run and ranged from 0.996 to 1.000 across three repeated runs. On the harder task of assigning MITRE ATT&CK tactics, only the LLM achieved positive agreement with the manual reference, although the agreement remained weak in absolute terms. The results show that rule-based signatures and a language model fail in complementary places and that combining them improves reliability on the evaluated benchmark, while broader generalization requires independent validation.



Keywords: PostgreSQL Honeypot, Cyber Deception, Threat Intelligence, MITRE ATT&CK, Large Language Model, Hybrid Intrusion Detection



DEDICATION

This thesis is dedicated to the future of cybersecurity and artificial intelligence in Indonesia, with the hope that this research contributes to a safer, more resilient digital landscape for the nation.



ACKNOWLEDGEMENT

I would like to express my deepest gratitude to Dr. Charles Lim, B.Sc., M.Sc. as my Thesis Advisor and Dr. Ir. Eka Budiarto, S.T., M.Sc. as my Co-Advisor, and to Mr. Yohanes Kloatubun as my mentor and discussion partner for the time, support, advice, and guidance given throughout this research project and the completion of this thesis report. It is because of their priceless contributions that this thesis report and the whole research project can arrive at this point.

I would like to thank my wife and daughter for the countless moral supports throughout my pursue of knowledge in SGU. I can't thank you enough for the time that already taken from you for me to make this thesis done.

I would like to thank all of my direct supervisor, my team at work and my friends especially Batch 17 MIT SGU for their companionship, and to the countless number of people who have helped me throughout this research project, either directly or indirectly.

Table of Contents

STATEMENT BY THE AUTHOR	2
RELATED PUBLICATION	3
ABSTRACT	4
DEDICATION	7
ACKNOWLEDGEMENT	8
TABLE OF CONTENTS	20
LIST OF FIGURES	23
LIST OF TABLES	26
1. INTRODUCTION	27
1.1 Background	27
1.2 Problem Statement	30
1.3 Research Objectives	31
1.4 Research Questions	32
1.5 Hypothesis	32
1.6 Scope and Limitation	33
1.6.1 Scope	33
1.6.2 Limitation	33
1.7 Research Timeline	33

1.8	Significance of Study	36
1.9	Thesis Structure	36
2.	LITERATURE REVIEW	38
2.1	The Business Imperative of Digital Transformation	38
2.1.1	Drivers and Urgency of Digital Transformation	38
2.1.2	Critical Success Factors for Digital Transformation	39
2.1.3	People Success Factor	39
2.1.4	Organization Success Factor	40
2.1.5	Technology Success Factor	41
2.1.6	Relational Component of Digital Transformation Success	41
2.1.7	Key Technology Interdependencies and Enabling Factors	42
2.1.8	The Critical Role of Data Infrastructure	42
2.1.9	Database Urgency in Digital Business Operations	43
2.2	Database Technology and PostgreSQL Positioning	44
2.2.1	The Role of Database Technology	46
2.2.2	Data Models and Workload Patterns	48
2.2.2.1	Online Transaction Processing Workloads	48
2.2.2.2	Online Analytical Processing Workloads	49
2.2.2.3	Comparison of OLTP and OLAP Workloads	50
2.2.2.4	Relationship Between Data Models and Workloads	51
2.2.3	Database Types and Polyglot Persistence	51
2.2.4	Relational and Non-Relational Databases in Modern Architectures	52
2.2.5	Why RDBMS Remains Central for Mission-Critical Workloads	53
2.2.6	Evolution of Enterprise RDBMS Platforms	54
2.2.7	Commercial RDBMS and the Historical Position of Oracle	54
2.2.8	PostgreSQL as an Enterprise-Relevant Open-Source RDBMS	55
2.2.9	Enterprise Business Needs Addressed by PostgreSQL	56
2.2.10	PostgreSQL Extensibility and Security Implications	58
2.2.11	Expanded PostgreSQL Positioning for This Thesis	58

2.2.11.1	PostgreSQL as Data Infrastructure	59
2.2.11.2	PostgreSQL as an Enterprise-Relevant RDBMS	60
2.2.11.3	PostgreSQL as a Security Research Object	61
2.2.11.4	What PostgreSQL Can Answer and What It Cannot Answer Alone	61
2.2.11.5	Why This Thesis Focuses on PostgreSQL Instead of Database Security in General	62
2.2.12	RDBMS Relevance and PostgreSQL Comparison Narrative	62
2.2.12.1	Commercial RDBMS Dominance and the Rise of PostgreSQL	63
2.2.12.2	SQL and NoSQL Comparison in Thesis Context	63
2.2.12.3	PostgreSQL Evolution From Inception Until Today	64
2.3	Database Security and the PostgreSQL Threat Landscape	65
2.3.1	Databases as Targets in Real-World Attacks	65
2.3.2	Common Attack Paths: Injection and Credential Abuse	65
2.3.3	PostgreSQL-Specific Threat Landscape	66
2.3.4	Common PostgreSQL Attack Trends Identified in Literature	67
2.3.5	Conventional Defense Mechanisms and Strategies	68
2.3.6	Emerging Strategies: Proactive Detection and Deception	69
2.3.7	Limits of Conventional Defenses and the Detection Problem	69
2.3.7.1	Signature- and Feature-Based Intrusion Detection	70
2.3.7.2	User and Entity Behavior Analytics	70
2.3.7.3	Perimeter and Network-Layer Defenses	71
2.3.7.4	Application-Layer Deception Beyond Honeypots	72
2.3.8	Proactive Defense via Deception	72
2.3.9	Threats to PostgreSQL: Attack Surface and Exposure	73
2.3.9.1	Exposure of Public-Facing Database Services	73
2.3.9.2	SQL-Layer Abuse: From Enumeration to Collection	73
2.3.9.3	Extensibility and Privileged Features as an Attack Surface	74
2.3.10	Integrated PostgreSQL Attack-Surface Synthesis	74

2.3.11	Detection Gap Reframed for PostgreSQL Honeypot Research	75
2.3.11.1	Why Logs Are Not Enough	75
2.3.11.2	Why Context Is Required	76
2.3.11.3	Why Deception Helps the Detection Gap	76
2.4	Cyber threat and threat modeling	77
2.4.1	Threat Modeling Before Cyber Deception	78
2.4.2	Cyber Resilience as the Defensive Context	78
2.4.3	Threat Landscape Analysis for PostgreSQL Honeypot Modeling	79
2.4.3.1	Threat Categories	79
2.4.3.2	Threat Behaviors and Tactics	79
2.4.3.3	Threat Actors and Attribution Boundary	80
2.4.3.4	Threat Mapping Using ATT&CK, CAPEC, CWE, and CVE	80
2.4.4	Operationalizing Threat Modeling for PostgreSQL Honeypot Analysis . .	80
2.4.5	Deep Justification of Threat Modeling, ATT&CK, CAPEC, and Profiling	81
2.4.5.1	Why Threat Modeling Is Required in a PostgreSQL Honeypot	81
2.4.5.2	Why MITRE ATT&CK Is Selected as the Primary Behavior-Mapping Layer	83
2.4.5.3	Why ATT&CK Alone Is Not Enough	84
2.4.5.4	Why CAPEC Is Required	85
2.4.5.5	Why CVE and CWE Are Kept as a Validation Layer	86
2.4.5.6	Why Threat Profiling Is Used Instead of Strong Attribution	87
2.4.5.7	Why Evidence-Based Confidence Is Required	89
2.4.5.8	Why the Framework Uses Layered Mapping Instead of One-Step Classification	90
2.4.5.9	Mapping Decision Rules for PostgreSQL Honeypot Profiling	90
2.4.5.10	Comparison of Threat Modeling and Knowledge Frameworks	92
2.4.5.11	Threat Profiling Output Structure	93
2.4.6	The PostgreSQL Vulnerability Landscape	94
2.4.7	Severity versus Exploitation Likelihood: CVSS, EPSS, and KEV	95

2.4.8	Evidence Taxonomy for PostgreSQL Honeypot Telemetry	96
2.4.8.1	Threat Categories Relevant to PostgreSQL Honeypot Telemetry	96
2.4.8.2	Threat Behaviors, Tactics, and Attack Patterns	97
2.4.8.3	Threat Actor Categories and Attribution Boundary	98
2.5	Cyber Deception for Database Security	99
2.5.1	Definition and Concept of Cyber Deception	99
2.5.2	Types of Cyber Deception	100
2.5.3	Cyber Deception for Database Security	103
2.5.4	Positioning the Honeypot as a Deception Mechanism	103
2.5.5	Deception for Databases: Positioning Honeypots Among Complementary Methods	104
2.5.6	Behavioral Deception and Attacker Engagement	105
2.5.6.1	Behavioral Profiling from Interaction Traces	105
2.5.7	Deception Development Philosophy for PostgreSQL Honeypot	106
2.6	PostgreSQL Honeypots and Deceptiveness Gaps	107
2.6.1	Concept and Modern Classification	108
2.6.2	Operational Challenges for High-Interaction Systems	108
2.6.3	Cyber Deception Honeypot for Database Systems	109
2.6.4	Implementation Challenges and Technology Requirements	110
2.6.5	Deceptiveness, Realism, and Current Honeypot Gaps	110
2.6.5.1	Protocol and Behavioral Fidelity Gap	111
2.6.5.2	Interaction Depth and State Continuity Gap	112
2.6.5.3	Synthetic Asset Realism Gap	112
2.6.5.4	Evidence Structure and Session Reconstruction Gap	113
2.6.5.5	Safety, Containment, and Reproducibility Gap	113
2.6.6	Honeypots for Relational Databases: Interaction, Fidelity, and Safety .	115
2.6.7	Honeypot Deployment for Database-Specific Protocols	116
2.6.8	Alternative Defensive Techniques and the Rationale for Selecting Honeypots	117

2.6.8.1	Established Techniques and Their Gaps	117
2.6.8.2	Why the Urgency Now	118
2.6.8.3	Why a PostgreSQL Honeypot: Research and Operational Rationale	119
2.6.8.4	Docker as a Foundation for High-Interaction Honeybots . . .	120
2.6.8.5	Architectural Baseline and Proposed Enhancement	122
2.7	Containerization and Safe High-Interaction Deployment	124
2.7.1	Containerization for Safe High-Interaction Honeybot Deployment . . .	124
2.7.2	Docker as an Isolation, Reset, and Reproducibility Mechanism	125
2.7.3	Why Containers Fit Deception Architectures	126
2.7.4	Security Requirements for Containerized Honeybots	126
2.7.5	Controls and Architecture for Safe High-Interaction Deployment	127
2.8	Behavior Mapping Analysis	128
2.8.1	Manual Behavior Mapping	128
2.8.2	From Manual Behavior Mapping to Automated Analysis	129
2.8.2.1	Vector Embeddings and Similarity Screening	129
2.8.3	Large Language Models: Definition and Capabilities	132
2.8.4	LLM for Behavior Analysis	133
2.8.5	LLM for Mapping and Report Assistance	133
2.8.6	Large Language Models for Adaptive Deception in Database Systems .	134
2.8.7	Bounding, Limitations, and Validation of the LLM	134
2.8.8	Evaluating the LLM Contribution	135
2.8.9	Selection of the Language Model	136
2.9	Threat Intelligence, Mapping, and Data Analysis Framework	138
2.9.1	Collection Phase: Attacker Interaction and Session Capture	139
2.9.2	Processing Phase: Data Processing and Behavior Classification	139
2.9.3	Mapping and Validation Phase: ATT&CK, CAPEC, and CVE/CWE . .	140
2.9.4	Output Phase: Threat Intelligence Output	141
2.9.5	Comparison of Framework Components and Alternatives	142

2.9.6	Overall Justification of the Proposed Framework	144
2.9.7	From Raw Events to Threat Intelligence: ATT&CK-Aligned Analysis .	145
2.9.8	Security Observability and the Honeypot Data Pipeline	145
2.9.9	Literature Basis for the Data Analysis Methodology	147
2.9.9.1	Data Collection: Capturing Session-Granular Interactions . .	147
2.9.9.2	Data Preprocessing: Normalization and Session Reconstruction	148
2.9.9.3	Data Enrichment: Adding External and Behavioral Context .	149
2.9.9.4	Data Transformation: Making Records Comparable and Queryable	149
2.9.9.5	Data Mining: Discovering Behavioral Structure from Sessions	150
2.9.9.6	Pattern Evaluation: Tactic-Aware Interpretation and Reporting	150
2.9.9.7	Threat Intelligence Output from Data Mining	151
2.10	Hybrid Intrusion Detection and Ensemble-Based Analysis	153
2.10.1	Signature-Based Detection: Deterministic Heuristics	153
2.10.2	Anomaly-Based Detection	156
2.10.3	Machine Learning	156
2.10.4	Hybrid Intrusion Detection	158
2.10.5	Ensemble Learning	158
2.10.6	Hard Voting versus Soft Voting	159
2.10.7	Level of Fusion and Score Normalization	161
2.10.8	Score Fusion	162
2.10.9	Threshold Selection and the Precision–Recall Complementarity	165
2.10.10	Origin and Robustness of the Decision Thresholds	165
2.11	Framework Evolution Across the Three Development Phases	167
2.11.1	Semester 1 (V1): The Baseline Framework	170
2.11.2	Semester 2 (V2): The Session-Level Framework	171
2.11.3	Semester 3 (V3): The Final Evidence-Mapping Framework	174
2.12	Related Works, Research Gap, and Methodological Synthesis	177
2.12.1	Thematic Literature Counter-Position	177

2.12.2	Synthesis of Related Honeypot Approaches	178
2.12.2.1	Synthesis of Related Works and Thesis Counter-Position . .	183
2.12.3	Integrated Related-Work Synthesis	184
2.12.4	Research Gap and Methodological Implications	186
2.13	Chapter Summary	187
2.14	Application-Facing Database Exposure	190
3.	Research Methodology	192
3.1	Methodology Overview	192
3.1.1	Alignment with the Implementation Phases	193
3.2	Data Analysis Pipeline	194
3.2.1	Attacker Interaction	195
3.2.2	Session Capture	196
3.2.3	Data Processing	196
3.2.4	Behavior Classification	197
3.2.5	Purpose of Classification and Construction of the Ground Truth	198
3.2.6	MITRE ATT&CK Mapping	199
3.2.7	CAPEC Mapping	200
3.2.8	CVE/CWE Validation	201
3.2.9	Threat Intelligence Output	201
3.3	Evidence Structure	202
3.4	Decision and Validation Logic	203
3.5	Threat Profiling Logic	204
3.6	Experimental Design: Static and Dynamic Treatments and Their Fusion	205
3.7	Success Criteria	206
3.8	Evaluation of the Framework	208
3.9	Safety and Ethical Considerations	209
3.10	Chapter Summary	210
4.	Technical Implementation	211

4.1	Implementation Overview	211
4.1.1	Phase 1	213
4.1.2	Phase 2	213
4.1.3	Phase 3	214
4.2	Hardware and Software Specification	215
4.3	Development of Deceptive Interaction Capability	218
4.3.1	Baseline Limitation of the Existing Honeypot	218
4.3.2	Deceptiveness Development Criteria	218
4.3.3	Implemented Deceptive Interaction Features	219
4.3.4	Phase Overview	220
4.4	Phase 1 Implementation	220
4.4.1	Phase 1 Objective	221
4.4.2	Phase 1 Technical Scope	221
4.4.3	Phase 1 Components	221
4.4.4	Phase 1 Event Capture Flow	222
4.4.5	Phase 1 Event Fields	222
4.4.6	Phase 1 Evidence Record Example	223
4.4.7	Phase 1 Output	223
4.4.8	Phase 1 Gap	223
4.5	Phase 2 Implementation	224
4.5.1	Phase 2 Objective	224
4.5.2	Phase 2 Technical Scope	224
4.5.3	Phase 2 Controlled Backend Design	224
4.5.4	Phase 2 Synthetic Asset Design	225
4.5.5	Phase 2 Backend Initialization Design	225
4.5.6	Phase 2 Proxy Mediation Flow	226
4.5.7	Phase 2 Session Reconstruction	227
4.5.8	Phase 2 Session Features	227
4.5.9	Phase 2 Output	227

4.5.10	Phase 2 Gap	228
4.6	Phase 3 Implementation	228
4.6.1	Phase 3 Objective	228
4.6.2	Phase 3 Technical Scope	229
4.6.3	Phase 3 Behavior Classification	229
4.6.4	Rule-Layer Severity Scoring and Weight Derivation	229
4.6.5	Phase 3 Asset Extraction	231
4.6.6	Phase 3 Mapping Logic	232
4.6.7	Phase 3 Validation Status	232
4.6.8	Phase 3 Multi LLM Revision	233
4.6.9	Phase 3 Output	233
4.7	Gap Relationship Between Phases	233
4.8	System Architecture Detail	234
4.8.1	Public Sensor Environment	234
4.8.2	Private SOC Environment	235
4.8.3	Reason for Separation	235
4.8.4	End-to-End Analytical Pipeline	235
4.9	Evidence Schema Detail	236
4.9.1	Session Record	238
4.9.2	Event Record	239
4.9.3	Query Record	239
4.9.4	Mapping Record	239
4.10	Implementation Algorithms	240
4.10.1	Event Capture Algorithm	240
4.10.2	Session Reconstruction Algorithm	240
4.10.3	Asset Extraction Algorithm	241
4.10.4	Evidence Based Mapping Algorithm	241
4.10.5	Report Generation Algorithm	242
4.11	Report Structure	242

4.11.1	Attacker Behavior Section	242
4.11.2	Affected Asset Section	243
4.11.3	Mapping Evidence Section	243
4.11.4	Actionable Priority Section	243
4.12	Evaluation Strategy	243
4.12.1	Technical Evaluation	243
4.12.2	Analytical Evaluation	244
4.13	Detailed Module Implementation Notes	244
4.13.1	Module 1: PostgreSQL Proxy Module	244
4.13.2	Module 2: Controlled Backend Module	244
4.13.3	Module 3: Local Evidence Module	245
4.13.4	Module 4: Transfer Module	245
4.13.5	Module 5: Session Engine Module	245
4.13.6	Module 6: Query Normalizer Module	246
4.13.7	Module 7: Asset Extractor Module	246
4.13.8	Module 8: Mapping Engine Module	246
4.13.9	Module 9: LLM Revision Module	246
4.13.10	Module 10: Report Generator Module	247
4.14	Phase Level Implementation Checklists	247
4.14.1	Phase 1 Checklist: Baseline Telemetry	247
4.14.2	Phase 2 Checklist: High Interaction Session Visibility	247
4.14.3	Phase 3 Checklist: Attacker to Asset Exposure Mapping	248
4.15	Acceptance Criteria Per Phase	248
4.15.1	Phase 1 Acceptance Criteria	248
4.15.2	Phase 2 Acceptance Criteria	248
4.15.3	Phase 3 Acceptance Criteria	249
4.16	Empirical Data Analysis and Findings	249
4.16.1	Quantitative Evaluation: Dataset, Ground Truth, and Results	251
4.16.2	Mapping Evaluation: MITRE ATT&CK Tactic Assignment	257

4.16.3	In-Context Rulebook Injection: a Mapping–Precision Trade-off	259
4.16.4	Threats to Validity	261
4.16.5	Discussion	262
4.16.6	Summary of the Quantitative Evaluation	263
4.16.7	Version 1 Findings: Medium-Interaction Honeypot (2025)	264
4.16.8	Version 2 Findings: Deterministic Regex-Only Detection (No LLM) . .	265
4.16.9	Phase 1 Findings (Version 3): Baseline Attacker Behavior Profile . . .	266
4.16.10	Phase 2 Findings (Version 3): Session-Level Behavior and Asset Ex- posure	267
4.16.11	Phase 3 Findings (Version 3): Evidence-Based Attacker to Asset Mapping	268
4.16.12	Diversity of Captured Attacks: Beyond a Single Finding	269
4.16.13	Result of Increasing Honeypot Interactivity	270
4.16.14	Comparative Evaluation: Version 2 (Published) vs. Version 3 (Final Enhancement)	271
4.17	Threats to Validity	273
4.18	Reproducibility Controls	274
4.19	Chapter Summary	274
4.20	Phase 3 Part 1: Original Direct-Database Experiment	275
4.21	Phase 3 Part 2: Extended Web-Application-Facing Experiment	275
4.21.1	Implementation Overview Addendum	275
4.21.2	Public-Network Attack Runs and Provenance	277
4.21.3	Aggregated Results	277
4.21.4	Evidence-Bounded Mapping of Strong Behaviors	279
4.21.5	Candidate Endpoint-Pattern Groups for Analyst Triage	281
4.21.6	What Was and Was Not Achieved	283
4.21.7	Extended Finding	283
5.	CONCLUSIONS AND FUTURE WORK	285
5.1	Conclusions	285
5.2	Contributions	286

5.3	Limitations	287
5.4	Future Work	287
A.	Complete PostgreSQL CVE Inventory	289
B.	Web Interface: SOC Intelligence Dashboard	294
C.	Web Decoy Interface Evidence	297
C.1	Raw JSON Evidence for Accepted Part 2 Mappings	297
D.	Supplementary Evidence Handling and Reproducibility	301
D.1	Evidence Boundary Between Experimental Parts	301
D.2	Verification Procedure	302
D.3	Interpretive Safeguards	303
REFERENCES		304
E.	Curriculum Vitae	317

