

Contents

Part I Introduction and Background

1	Introduction	3
1.1	Organization and Structure	7
2	Related Work on Secure Deletion	11
2.1	Introduction	11
2.2	Related Work	11
2.2.1	Layers and Interfaces	12
2.2.2	Physical-Layer and Controller-Layer Sanitization	15
2.2.3	User-Level Solutions	17
2.2.4	File-System-Level Solutions with In-Place Updates	20
2.2.5	Cross-layer Solutions	22
2.2.6	Summary	23
2.3	Adversarial Model	23
2.3.1	Classes of Adversarial Capabilities	23
2.3.2	Summary	25
2.4	Analysis of Solutions	26
2.4.1	Classes of Environmental Assumptions	27
2.4.2	Classes of Behavioural Properties	27
2.4.3	Summary	30
3	System Model and Security Goal	33
3.1	Introduction	33
3.2	System Model	33
3.3	Storage Medium Models	34
3.4	Adversarial Model	36
3.5	Security Goal	37

Part II Secure Deletion for Mobile Storage

4	Flash Memory: Background and Related Work	47
4.1	Overview	47
4.2	Flash Memory	48
4.2.1	In-Place Updates and Log-Structured File Systems	49
4.2.2	Flash Translation Layer	51
4.2.3	Flash File Systems	52
4.2.4	Generalizations to Other Media	52
4.3	Related Work for Flash Secure Deletion	52
4.4	Summary	55
5	User-Level Secure Deletion on Log-Structured File Systems	57
5.1	Introduction	57
5.2	System and Adversarial Model	58
5.3	YAFFS	58
5.4	Data Deletion in Existing Log-Structured File Systems	59
5.4.1	Instrumented YAFFS	60
5.4.2	Simulating Larger Storage Media	61
5.5	User-Space Secure Deletion	63
5.5.1	Purging	64
5.5.2	Ballooning	65
5.5.3	Hybrid Solution: Ballooning with Purging	67
5.6	Experimental Evaluation	67
5.6.1	Experimental Results	68
5.7	Summary	72
5.8	Research Questions	72
6	Data Node Encrypted File System	73
6.1	Introduction	73
6.2	System and Adversarial Model	73
6.3	DNEFS's Design	74
6.3.1	Key Storage Area	75
6.3.2	Keystore	75
6.3.3	Clocked Keystore Implementation	77
6.3.4	Clock Operation: KSA Update	79
6.3.5	Key-State Map	79
6.3.6	Summary	81
6.4	Extensions and Optimizations	81
6.4.1	Granularity Trade-off	81
6.4.2	KSA Update Policies	83
6.4.3	KSA Organization	83
6.4.4	Improving Reliability	83
6.4.5	Encrypted File System	84
6.5	Summary	84

- 6.6 Research Questions 85
- 7 UBIFSec: Adding DNEFS to UBIFS 87**
 - 7.1 Introduction 87
 - 7.2 System and Adversarial Model 87
 - 7.3 Background 87
 - 7.3.1 MTD and UBI Layers 88
 - 7.3.2 UBIFS 88
 - 7.4 UBIFSec Design 90
 - 7.4.1 Key Storage Area 90
 - 7.4.2 Key-State Map 92
 - 7.4.3 Summary 94
 - 7.5 Experimental Validation 94
 - 7.5.1 Android Implementation 96
 - 7.5.2 Wear Analysis 96
 - 7.5.3 Power Consumption 98
 - 7.5.4 Throughput Analysis 98
 - 7.5.5 Timing Analysis 99
 - 7.6 Conclusions 101
 - 7.7 Practitioner’s Notes 101

Part III Secure Deletion for Remote Storage

- 8 Cloud Storage: Background and Related Work 105**
 - 8.1 Introduction 105
 - 8.2 Persistent Storage 105
 - 8.2.1 Securely Deleting and Persistent Combination 106
 - 8.2.2 Cloud Storage 106
 - 8.3 Related Work 108
 - 8.4 Summary 113
- 9 Secure Data Deletion from Persistent Media 115**
 - 9.1 Introduction 115
 - 9.2 System and Adversarial Model 116
 - 9.3 Graph Theory Background 116
 - 9.4 Graph-Theoretic Model of Key Disclosure 118
 - 9.4.1 Key Disclosure Graph 118
 - 9.4.2 Secure Deletion 120
 - 9.5 Shadowing Graph Mutations 120
 - 9.5.1 Mangrove Preservation 123
 - 9.5.2 Shadowing Graph Mutation Chains 125
 - 9.5.3 Mangrove Key Disclosure Graphs in Related Work 126
 - 9.6 Summary 127
 - 9.7 Research Questions 128

- 10 B-Tree-Based Secure Deletion** 129
 - 10.1 Introduction 129
 - 10.2 System and Adversarial Model 130
 - 10.3 Background 130
 - 10.3.1 B-Tree Storage Operations 130
 - 10.3.2 B-Tree Balance Operations 131
 - 10.4 Securely Deleting B-Tree Design 131
 - 10.4.1 Cryptographic Details 132
 - 10.4.2 Data Integrity 132
 - 10.4.3 Versioning 133
 - 10.4.4 Skeleton Tree 133
 - 10.4.5 Commitment 135
 - 10.4.6 Crash Safety 135
 - 10.5 Implementation Details 136
 - 10.5.1 Data Storage 136
 - 10.5.2 Network Block Device 137
 - 10.5.3 Virtual Storage Device 137
 - 10.5.4 Caches 137
 - 10.6 Experimental Evaluation 138
 - 10.6.1 Workloads 138
 - 10.6.2 Caching 139
 - 10.6.3 B-Tree Properties 140
 - 10.7 Conclusions 140
 - 10.8 Practitioner’s Notes 141

- 11 Robust Key Management for Secure Data Deletion** 143
 - 11.1 Introduction 143
 - 11.2 System and Adversarial Model 144
 - 11.2.1 System Entities 145
 - 11.2.2 Adversarial Model 146
 - 11.3 Distributed Keystore 147
 - 11.3.1 Distributed Clocked Keystore 147
 - 11.3.2 Distributed Keystore Correctness 148
 - 11.4 Synchronization 150
 - 11.5 Byzantine Robustness 153
 - 11.6 Keystore Secure Deletion 156
 - 11.6.1 Key Pools and Encryption Keys 157
 - 11.6.2 Encryption Key Encoding 159
 - 11.6.3 XOR-Based Encoding 159
 - 11.6.4 Security Analysis 162
 - 11.7 Implementation Details 166
 - 11.8 Experimental Validation 170
 - 11.9 Conclusions 173
 - 11.10 Research Questions 173
 - 11.11 Practitioner’s Notes 174

Part IV Conclusions

- 12 Conclusion and Future Work** 177
 - 12.1 Summary of Contributions 177
 - 12.2 Related and Complementary Research 177
 - 12.2.1 Information Deletion 179
 - 12.2.2 File Carving 180
 - 12.2.3 Steganographic and Deniable Storage 180
 - 12.2.4 History Independence 181
 - 12.2.5 Provable Deletion 182
 - 12.3 Future Work 182
 - 12.3.1 New Types of Storage Media 182
 - 12.3.2 Benchmarks for Different Storage 183
 - 12.3.3 Secure-Deletion Data Structure Selection 183
 - 12.3.4 Formalization 184
 - 12.3.5 DNEFS for FTLs 184
 - 12.4 Concluding Remarks 185
- References** 187
- Glossary** 193
- Index** 201



<http://www.springer.com/978-3-319-28777-5>

Secure Data Deletion

Reardon, J.

2016, XVII, 203 p. 32 illus., Hardcover

ISBN: 978-3-319-28777-5