CONTENTS IN DETAIL

FOREWORD **xv**

ACKNOWLEDGMENTS **xvii**

INTRODUCTION **xix**

Who This Book Is For xx
Who This Book Is Not For xx
How This Book Is Organized xx

PART I: RED TEAMING FUNDAMENTALS 1

1
PRINCIPLES OF APPLICATION DESIGN AND DEVELOPMENT 3

The Three Rs of Application Development 4
Robustness and Requirements Gathering 4
Reusability and Modular Coding 6
Reliability and Exception Handling 8
Other Application Development Best Practices 10
Adopting Naming Standards and Syntax Conventions 10
Adding a Logger 10
Designing Redirection Infrastructure 11
Securing Your Infrastructure 12
Runtime Requirements 13
Choosing Between Tool Modification and Development 15
The Benefits of Multistage Builds 16
Summary 19

2
EVASION STRATEGIES 21

Detection Engineering Patterns 22
SOC Patterns 22
Estimating Resources 23
Optimizing the Timing of Your Operations 24
Blending In 25
Using Low-Entropy Encryption 25
Avoiding Risky Binaries and Packages 28
Following Naming Conventions 30
Padding Files 30
Adding Code Signing 31

Switching Languages and Architecture 31
Other Evasion Techniques 32
Mimicking End User Behavior 33
Distracting Defenders with Tandem Operations 33
Using False Flags for Misdirection 33
Summary 34

PART II: HANDS-ON EVASIVE TOOL DEVELOPMENT 35

3
ENUMERATING WITH TRAFFIC REDIRECTION 37

Functional Design 38
Technical Requirements 38
Deployment Architecture 38
Build a Web Scanner with AWS Lambda 40
Getting Started with AWS Lambda 40
Creating a Python Function to Capture Web Responses 42
Creating Persistent Scanning Infrastructure 45
Getting Started with Amazon VPC 45
Building a Reusable Deployment Template with CloudFormation 46
Implementing the Serverless Scanner with Step Functions 50
Deploying the CloudFormation Solution 55
Deploying the AWS Lambda Solution 61
Switching the Source IP Addresses 63
Tearing Down Your Scanning Environment 64
Summary 64

4
DEVELOPING COMMAND-AND-CONTROL IMPLANTS 65

Functional Design 66
Technical Requirements 66
Deployment Architecture 66
Creating a Simple Command-and-Control Prototype 68
Preparing Your GCP Environment 68
Setting Up Go Modules and Dependencies 75
Writing the Base Implant Code 76
Improving the Implant's Usability and Stealth 82
Configuring Firestore Permissions 83
Embedding Credentials and Adding Firestore 84
Creating the Firestore Database 90
Testing and Executing the Implant 91
Advanced Functionality 93
Summary 96

5
CREATING LATERAL EXPLOITS WITH WORMS 97

Functional Design 98
Technical Requirements 98
Deployment Architecture 98
Setting Up an Infection Instance 99
Configuring GCP Cloud Shell and Terraform 99
Deploying the Test Infrastructure 100
Preparing the Worm Code 106
Compiling and Encoding the Implant Code 111
Setting Up the Infection 112
Importing the Precompiled Binary 112
Manually Inserting the Implant 112
Enabling SSH Password Authentication 113
Infecting the First Host 114
Summary 115

6
ENUMERATING LOCALLY WITHOUT LOLBINS 117

Functional Requirements 118
What Are LOLBins? 118
Why to Avoid LOLBins 118
The Top Five Enumeration Categories 119
Operating System 119
Users and Groups 119
Hostname 120
Network and Firewall Configuration 120
Running Processes, Services, and Scheduled Tasks 120
Replacing LOLBins with Your Own Tools 121
Detecting and Reporting the Operating System 122
Querying Active Directory for Users and Groups 123
Extracting and Displaying the Hostname 127
Returning Network and Firewall Configuration Details 128
Identifying Running Processes, Services, and Scheduled Tasks 135
Summary 136

7
BYPASSING DETECTION WITH HYBRID PACKING 137

Functional Requirements 138
Tooling and Process Breakdown 138
Transferring Executables into Stealthy Shellcode Using Ex2Shell 139
Hiding Your Payload in Plain Sight with Encode 142
Executing the Payload in Memory with the Loader Tool 150
Summary 157

8
STAGING AND EXFILTRATING DATA COVERTLY 159

Functional Design . . . 160
Staging . . . 160
Exfiltration . . . 160
Retrieval . . . 160
Technical Requirements . . . 161
Deployment Architecture . . . 161
Setting Up Your GCP Environment . . . 162
Building the CAB Staging Tool . . . 164
Developing the Exfiltration Client . . . 169
Creating the File Retrieval Client . . . 172
Executing the Full Exfiltration Pipeline . . . 175
Summary . . . 178

PART III: TESTING AND VALIDATION 179

9
BUILDING DETECTION TOOLS 181

Functional Design . . . 182
Binary Property Analysis . . . 182
Network Behavior Analysis . . . 182
Technical Requirements . . . 182
Deployment Architecture . . . 183
Building the Go Binary Analyzer . . . 183
Building the Network Traffic Analyzer in Python . . . 188
Generating the PAN Traffic Log . . . 188
Monitoring for Suspicious Traffic . . . 192
Summary . . . 200

10
EXECUTING CONTROLLED REVEALS 201

Functional Design . . . 202
Technical Requirements . . . 202
Deployment Architecture . . . 202
Simulating Cloud Persistence with Lambda IAM Operations . . . 203
Replicating Ransomware TTPs on EC2 . . . 204
Setting Up AWS CloudTrail . . . 205
Automating Deployment with CloudFormation . . . 206
Setting Up and Executing Your Simulation Stack . . . 212
Building Detection Rules from CloudTrail and CloudWatch Events . . . 216
Summary . . . 219

APPENDIX: TECHNICAL PREREQUISITES 221

INDEX 231