1.

Record Nr.	UNINA9910822423003321
Autore	Smith Craig (Reverse engineer)
Titolo	The car hacker's handbook : a guide for the penetration tester / / by Craig Smith
Pubbl/distr/stampa	San Francisco, [California] : , : No Starch Press, , 2016 ©2016
ISBN	1-4571-9884-3 1-59327-770-9
Edizione	[1st edition]
Descrizione fisica	1 online resource (306 pages) : illustrations
Disciplina	629.2/72
Soggetti	Automotive computers - Security measures Automobiles - Performance Automobiles - Customizing Penetration testing (Computer security) Automobiles - Vandalism - Prevention
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Intro -- Title Page -- Copyright Page -- About the Author -- About the Contributing Author -- About the Technical Reviewer -- Brief Contents -- Contents in Detail -- Foreword by Chris Evans -- Acknowledgments -- Introduction -- Why Car Hacking Is Good for All of Us -- What's in This Book -- Chapter 1: Understanding Threat Models -- Finding Attack Surfaces -- Threat Modeling -- Level 0: Bird's-Eye View -- Level 1: Receivers -- Level 2: Receiver Breakdown -- Threat Identification -- Level 0: Bird's-Eye View -- Level 1: Receivers -- Level 2: Receiver Breakdown -- Threat Rating Systems -- The DREAD Rating System -- CVSS: An Alternative to DREAD -- Working with Threat Model Results -- Summary -- Chapter 2: Bus Protocols -- The CAN Bus -- The OBD-II Connector -- Finding CAN Connections -- CAN Bus Packet Layout -- The ISO-TP Protocol -- The CANopen Protocol -- The GMLAN Bus -- The SAE J1850 Protocol -- The PWM Protocol -- The VPW Protocol -- The Keyword Protocol and ISO 9141-2 -- The Local Interconnect Network Protocol -- The MOST Protocol -- MOST Network Layers -- MOST Control Blocks -- Hacking MOST -- The FlexRay Bus --

Hardware -- Network Topology -- Implementation -- FlexRay Cycles -- Packet Layout -- Sniffing a FlexRay Network -- Automotive Ethernet -- OBD-II Connector Pinout Maps -- The OBD-III Standard -- Summary -- Chapter 3: Vehicle Communication With SocketCAN -- Setting Up can-utils to Connect to CAN Devices -- Installing can-utils -- Configuring Built-In Chipsets -- Configuring Serial CAN Devices -- Setting Up a Virtual CAN Network -- The CAN Utilities Suite -- Installing Additional Kernel Modules -- The can-isotp.ko Module -- Coding SocketCAN Applications -- Connecting to the CAN Socket -- Setting Up the CAN Frame -- The Procfs Interface -- The Socketcand Daemon -- Kayak -- Summary -- Chapter 4: Diagnostics and Logging. Diagnostic Trouble Codes -- DTC Format -- Reading DTCs with Scan Tools -- Erasing DTCs -- Unified Diagnostic Services -- Sending Data with ISO-TP and CAN -- Understanding Modes and PIDs -- Brute-Forcing Diagnostic Modes -- Keeping a Vehicle in a Diagnostic State -- Event Data Recorder Logging -- Reading Data from the EDR -- The SAE J1698 Standard -- Other Data Retrieval Practices -- Automated Crash Notification Systems -- Malicious Intent -- Summary -- Chapter 5: Reverse Engineering the CAN Bus -- Locating the CAN Bus -- Reversing CAN Bus Communications with can-utils and Wireshark -- Using Wireshark -- Using candump -- Grouping Streamed Data from the CAN Bus -- Using Record and Playback -- Creative Packet Analysis -- Getting the Tachometer Reading -- Creating Background Noise with the Instrument Cluster Simulator -- Setting Up the ICSim -- Reading CAN Bus Traffic on the ICSim -- Changing the Difficulty of ICSim -- Reversing the CAN Bus with OpenXC -- Translating CAN Bus Messages -- Writing to the CAN Bus -- Hacking OpenXC -- Fuzzing the CAN Bus -- Troubleshooting When Things Go Wrong -- Summary -- Chapter 6: ECU Hacking -- Front Door Attacks -- J2534: The Standardized Vehicle Communication API -- Using J2534 Tools -- KWP2000 and Other Earlier Protocols -- Capitalizing on Front Door Approaches: Seed-Key Algorithms -- Backdoor Attacks -- Exploits -- Reversing Automotive Firmware -- Self-Diagnostic System -- Library Procedures -- Comparing Bytes to Identify Parameters -- Identifying ROM Data with WinOLS -- Code Analysis -- A Plain Disassembler at Work -- Interactive Disassemblers -- Summary -- Chapter 7: Building and Using ECU Test Benches -- The Basic ECU Test Bench -- Finding an ECU -- Dissecting the ECU Wiring -- Wiring Things Up -- Building a More Advanced Test Bench -- Simulating Sensor Signals -- Hall Effect Sensors -- Simulating Vehicle Speed.

Summary -- Chapter 8: Attacking ECUS And Other Embedded Systems -- Analyzing Circuit Boards -- Identifying Model Numbers -- Dissecting and Identifying a Chip -- Debugging Hardware with JTAG and Serial Wire Debug -- JTAG -- Serial Wire Debug -- The Advanced User Debugger -- Nexus -- Side-Channel Analysis with the ChipWhisperer -- Installing the Software -- Prepping the Victim Board -- Brute-Forcing Secure Boot Loaders in Power-Analysis Attacks -- Prepping Your Test with AVRDUDESS -- Setting Up the ChipWhisperer for Serial Communications -- Setting a Custom Password -- Resetting the AVR -- Setting Up the ChipWhisperer ADC -- Monitoring Power Usage on Password Entry -- Scripting the ChipWhisperer with Python -- Fault Injection -- Clock Glitching -- Setting a Trigger Line -- Power Glitching -- Invasive Fault Injection -- Summary -- Chapter 9: In-Vehicle Infotainment Systems -- Attack Surfaces -- Attacking Through the Update System -- Identifying Your System -- Determining the Update File Type -- Modifying the System -- Apps and Plugins -- Identifying Vulnerabilities -- Attacking the IVI Hardware -- Dissecting the IVI Unit's Connections -- Disassembling the IVI Unit --

Infotainment Test Benches -- GENIVI Meta-IVI -- Automotive Grade Linux -- Acquiring an OEM IVI for Testing -- Summary -- Chapter 10: Vehicle-to-Vehicle Communication -- Methods of V2V Communication -- The DSRC Protocol -- Features and Uses -- Roadside DSRC Systems -- WAVE Standard -- Tracking Vehicles with DSRC -- Security Concerns -- PKI-Based Security Measures -- Vehicle Certificates -- Anonymous Certificates -- Certificate Provisioning -- Updating the Certificate Revocation List -- Misbehavior Reports -- Summary -- Chapter 11: Weaponizing CAN Findings -- Writing the Exploit in C -- Converting to Assembly Code -- Converting Assembly to Shellcode -- Removing NULLs.

Creating a Metasploit Payload -- Determining Your Target Make -- Interactive Probing -- Passive CAN Bus Fingerprinting -- Responsible Exploitation -- Summary -- Chapter 12: Attacking Wireless Systems with SDR -- Wireless Systems and SDR -- Signal Modulation -- Hacking with TPMS -- Eavesdropping with a Radio Receiver -- TPMS Packets -- Activating a Signal -- Tracking a Vehicle -- Event Triggering -- Sending Forged Packets -- Attacking Key Fobs and Immobilizers -- Key Fob Hacks -- Attacking a PKES System -- Immobilizer Cryptography -- Physical Attacks on the Immobilizer System -- Flashback: Hotwiring -- Summary -- Chapter 13: Performance Tuning -- Performance Tuning Trade-Offs -- ECU Tuning -- Chip Tuning -- Flash Tuning -- Stand-Alone Engine Management -- Summary -- Appendix A: Tools of the Trade -- Hardware -- Lower-End CAN Devices -- Higher-End CAN Devices -- Software -- Wireshark -- PyOBD Module -- Linux Tools -- CANiBUS Server -- Kayak -- SavvyCAN -- O2OO Data Logger -- Caring Caribou -- c0f Fingerprinting Tool -- UDSim ECU Simulator -- Octane CAN Bus Sniffer -- AVRDUDESS GUI -- RomRaider ECU Tuner -- Komodo CAN Bus Sniffer -- Vehicle Spy -- Appendix B: Diagnostic Code Modes and PIDs -- Modes Above 0x10 -- Useful PIDs -- Appendix C: Creating Your Own Open Garage -- Filling Out the Character Sheet -- When to Meet -- Affiliations and Private Memberships -- Defining Your Meeting Space -- Contact Information -- Initial Managing Officers -- Equipment -- Abbreviations -- Index -- Footnotes -- Chapter 10: Vehicle-to-Vehicle Communication -- Chapter 12: Attacking Wireless Systems with SDR.

Sommario/riassunto

The Car Hacker's Handbook shows how to identify vulnerabilities in modern automotive vehicles.