

Table of Contents

| | |
|---|----|
| • Preface | 13 |
| Chapter 1 • Introduction | 14 |
| 1.1 • What is home automation? | 14 |
| 1.2 • Why use a Raspberry Pi as a home automation gateway? | 15 |
| 1.3 • The properties of a good home automation system. | 16 |
| 1.3.1 • Secure | 17 |
| 1.3.2 • Modular | 18 |
| 1.3.3 • Open-Source | 19 |
| 1.3.4 • Self-sufficient. | 20 |
| 1.4 • How to use this book. | 23 |
| 1.5 • Summary and further exploration | 25 |
| Chapter 2 • The Raspberry Pi as a home automation gateway | 27 |
| 2.1 • Which Raspberry Pi models are suitable for home automation? | 27 |
| 2.2 • Requirements for a reliable home automation gateway | 30 |
| 2.3 • Installing Raspberry Pi OS | 32 |
| 2.4 • Setting up network connectivity with Ethernet or Wi-Fi | 35 |
| 2.4.1 • Ethernet | 35 |
| 2.4.2 • Wi-Fi | 36 |
| 2.4.3 • Setting a fixed IP address | 36 |
| 2.5 • Remote access using SSH | 37 |
| 2.5.1 • Enabling the SSH server | 37 |
| 2.5.2 • Connecting with the SSH client | 37 |
| 2.6 • Basic setup | 39 |
| 2.7 • The tmux terminal multiplexer | 40 |
| 2.7.1 • The basics of tmux: windows | 40 |
| 2.7.2 • Working with tmux sessions | 41 |
| 2.7.3 • Seeing more at the same time with panes. | 42 |
| 2.7.4 • Copying and pasting text | 43 |
| 2.8 • Python | 43 |
| 2.8.1 • Virtual environments | 44 |
| 2.8.2 • Package requirements | 45 |
| 2.9 • Docker | 46 |
| 2.9.1 • Installing Docker | 46 |
| 2.9.2 • Installing Docker Compose | 48 |

| | |
|---|-----------|
| 2.9.3 • Creating a Docker Compose YAML file | 49 |
| 2.10 • Summary and further exploration | 51 |
| Chapter 3 • Secure your home automation system | 53 |
| 3.1 • Some general computer security principles | 53 |
| 3.2 • Isolate your home automation devices | 55 |
| 3.2.1 • Physical isolation | 55 |
| 3.2.2 • VLANs | 57 |
| 3.2.3 • Firewalls | 58 |
| 3.3 • User management | 62 |
| 3.3.1 • Permissions | 62 |
| 3.3.2 • Passwords | 63 |
| 3.3.3 • Lifecycle | 65 |
| 3.4 • Encryption | 65 |
| 3.4.1 • Your threat model | 65 |
| 3.4.2 • TLS | 66 |
| 3.4.3 • Setting up your own CA with mkcert | 67 |
| 3.4.4 • Creating a CA | 68 |
| 3.4.5 • Creating and signing a certificate | 70 |
| 3.4.6 • Keeping your root CA key secure | 71 |
| 3.5 • Keeping your software up-to-date | 72 |
| 3.5.1 • Update apt packages | 72 |
| 3.5.2 • Update Docker images | 76 |
| 3.5.3 • Update pip packages | 77 |
| 3.5.4 • Update manually installed packages | 77 |
| 3.5.5 • Update your home automation devices | 78 |
| 3.6 • Summary and further exploration | 78 |
| Chapter 4 • MQTT (Message Queuing Telemetry Transport) | 80 |
| 4.1 • What is MQTT? | 80 |
| 4.1.1 • Central intermediary | 80 |
| 4.1.2 • Hierarchical names | 81 |
| 4.1.3 • Using wildcards | 82 |
| 4.2 • Installing and configuring the Mosquitto MQTT broker | 83 |
| 4.2.1 • A basic Mosquitto setup | 83 |
| 4.2.2 • Testing your setup with the Mosquitto clients | 85 |

| | |
|---|-----|
| 4.2.3 • A secure Mosquitto setup | 86 |
| 4.2.4 • Testing your secure setup with the Mosquitto clients | 90 |
| 4.2.5 • Default options for Mosquitto clients | 92 |
| 4.3 • Using graphical MQTT clients | 93 |
| 4.3.1 • MQTT.fx | 93 |
| 4.3.2 • MQTT Explorer | 95 |
| 4.4 • Using MQTT in Python | 97 |
| 4.5 • Direct communication between other containers and Mosquitto | 100 |
| 4.6 • Summary and further exploration | 103 |
| Chapter 5 • TCP/IP | 105 |
| 5.1 • Wake other network devices | 105 |
| 5.2 • Remote control with SSH | 107 |
| 5.2.1 Run commands on other devices | 108 |
| 5.2.2 • Secure passwordless logins using SSH keys | 109 |
| 5.3 • Collecting information from devices using SNMP | 111 |
| 5.3.1 • Walking through the MIB tree | 111 |
| 5.3.2 • Collecting your router's version using SNMP | 113 |
| 5.3.3 • Collecting your printer's ink levels | 114 |
| 5.4 • Using devices with a HTTP/REST API | 116 |
| 5.4.1 • Setting up a Shelly device for secure remote control | 117 |
| 5.4.2 • Using Shelly's HTTP API with curl | 118 |
| 5.4.3 • Using the HTTP API in Python | 119 |
| 5.5 • Creating a video surveillance system | 121 |
| 5.5.1 • Turn your Raspberry Pi into an IP camera | 123 |
| 5.5.2 • Turn your Raspberry Pi into a camera controller | 125 |
| 5.5.3 • Viewing your remote cameras | 128 |
| 5.5.4 • Motion detection | 129 |
| 5.5.5 • Notifications on motion | 131 |
| 5.6 • Summary and further exploration | 133 |
| Chapter 6 • Bluetooth | 134 |
| 6.1 • An introduction to Bluetooth Low Energy | 134 |
| 6.1.1 • Broadcasting data | 134 |
| 6.1.2 • Connecting to services | 135 |
| 6.2 • Enabling Bluetooth | 137 |
| 6.3 • Investigating Bluetooth Low Energy devices | 138 |
| 6.3.1 • Scanning for Bluetooth Low Energy devices | 139 |

| | |
|---|------------|
| 6.3.2 • Dumping raw Bluetooth broadcast data | 140 |
| 6.3.3 • Discovering device characteristics | 141 |
| 6.3.4 • Reading device characteristics | 142 |
| 6.4 • Reading BLE sensor values in Python | 143 |
| 6.4.1 • RuuviTag Sensor. | 143 |
| 6.4.2 • Miflora | 146 |
| 6.5 • Relaying Bluetooth sensor values with bt-mqtt-gateway | 148 |
| 6.5.1 • Configuring bt-mqtt-gateway | 148 |
| 6.5.2 • Running bt-mqtt-gateway | 150 |
| 6.6 • Presence detection with Bluetooth | 152 |
| 6.6.1 • Presence detection with monitor.sh | 152 |
| 6.6.2 • Configuring and running monitor.sh | 153 |
| 6.6.3 • Trigger arrival and departure scans in monitor.sh | 155 |
| 6.7 • Summary and further exploration | 156 |
| Chapter 7 • 433.92 MHz | 157 |
| 7.1 • 433.92 MHz protocols | 157 |
| 7.2 • Hardware requirements | 158 |
| 7.2.1 • Receiver | 158 |
| 7.2.2 • Antenna | 159 |
| 7.3 • Receiving sensor values with rtl_433 | 160 |
| 7.3.1 • Installing rtl_433toMQTT | 161 |
| 7.3.2 • Configuring rtl_433 | 163 |
| 7.4 • Publishing 433.92 MHz sensor values to MQTT | 165 |
| 7.5 • Summary and further exploration | 166 |
| Chapter 8 • Z-Wave | 168 |
| 8.1 • An introduction to Z-Wave | 168 |
| 8.1.1 • The specification. | 168 |
| 8.1.2 • How does Z-Wave work? | 169 |
| 8.2 • Choosing a Z-Wave transceiver | 170 |
| 8.2.1 • Transceiver on the GPIO header: RaZberry | 171 |
| 8.2.2 • USB Transceiver | 172 |
| 8.3 • OpenZWave and Zwave2Mqtt | 173 |
| 8.3.1 • Installing Zwave2Mqtt | 174 |
| 8.3.2 • Configuring Zwave2Mqtt | 175 |
| 8.3.3 • Using the Zwave2Mqtt Control Panel | 179 |
| 8.4 • Using your Z-Wave devices with MQTT | 183 |

| | |
|--|------------|
| 8.4.1 • Reading sensor values | 184 |
| 8.4.2 • Controlling switches | 185 |
| 8.5 • Summary and further exploration | 186 |
| Chapter 9 • Zigbee | 188 |
| 9.1 • An introduction to Zigbee | 188 |
| 9.1.1 • The specification | 188 |
| 9.1.2 • How does Zigbee work? | 189 |
| 9.2 • Creating a Zigbee transceiver | 189 |
| 9.2.1 • Connect the downloader cable | 190 |
| 9.2.2 • Install the flashing software | 192 |
| 9.2.3 • Flash the firmware | 192 |
| 9.3 • Zigbee2mqtt and Zigbee2MqttAssistant | 193 |
| 9.3.1 • Connecting the CC2531 | 194 |
| 9.3.2 • Installing Zigbee2mqtt and Zigbee2MqttAssistant | 195 |
| 9.3.3 • Configuring Zigbee2mqtt and Zigbee2MqttAssistant | 196 |
| 9.3.4 • Using Zigbee2MqttAssistant | 198 |
| 9.4 • Using our Zigbee devices with MQTT | 200 |
| 9.4.1 • Reading sensor values | 201 |
| 9.4.2 • Controlling switches | 201 |
| 9.5 • Summary and further exploration | 202 |
| Chapter 10 • Automating your home | 203 |
| 10.1 • Node-RED | 204 |
| 10.1.1 • Installing Node-RED | 204 |
| 10.1.2 • Adding authentication to Node-RED | 205 |
| 10.1.3 • Using Node-RED over HTTPS | 207 |
| 10.1.4 • Creating Node-RED flows | 209 |
| 10.1.5 • Installing extra nodes in Node-RED | 213 |
| 10.1.6 • Creating a dashboard in Node-RED | 215 |
| 10.2 • Home Assistant | 219 |
| 10.2.1 • Installing Home Assistant | 219 |
| 10.2.2 • Integrating MQTT | 221 |
| 10.2.3 • Creating automation rules | 224 |
| 10.3 • AppDaemon | 226 |
| 10.3.1 • Installing AppDaemon | 226 |

| | |
|---|------------|
| 10.3.2 • Creating an AppDaemon app with MQTT: the time | 229 |
| 10.3.3 • Creating an AppDaemon app with MQTT: garage door alert | 231 |
| 10.4 • Summary and further exploration | 233 |
| Chapter 11 • Notifications | 234 |
| 11.1 • Forwarding local email | 234 |
| 11.1.1 • Installing Nullmailer | 234 |
| 11.1.2 • Testing Nullmailer | 236 |
| 11.1.3 • Using Nullmailer | 237 |
| 11.2 • Forwarding emails from Docker containers | 237 |
| 11.2.1 • Installing docker-postfix | 237 |
| 11.2.2 • Sending emails to docker-postfix | 239 |
| 11.3 • Push notifications with Gotify | 241 |
| 11.3.1 • Installing the Gotify server | 242 |
| 11.3.2 • Adding applications to Gotify | 243 |
| 11.3.3 • Using Gotify applications | 244 |
| 11.3.4 • Using Gotify clients | 247 |
| 11.4 • Notifications on receiving MQTT messages | 248 |
| 11.4.1 • Installing mqttwarn | 248 |
| 11.4.2 • Sending emails with mqttwarn | 251 |
| 11.4.3 • Transforming and filtering payloads | 252 |
| 11.5 • Summary and further exploration | 255 |
| Chapter 12 • Voice control | 256 |
| 12.1 • A basic Rhasspy setup | 256 |
| 12.1.1 • Hardware requirements | 257 |
| 12.1.2 • Configure audio hardware | 257 |
| 12.1.3 • Installing Rhasspy | 260 |
| 12.1.4 • Rhasspy's settings | 261 |
| 12.1.5 • Configuring audio | 262 |
| 12.1.6 • Configuring the wake word | 263 |
| 12.1.7 • Configuring text to speech | 263 |
| 12.1.8 • Configuring speech to text | 264 |
| 12.1.9 • Configuring intent recognition | 265 |
| 12.1.10 • Configuring dialogue management | 266 |
| 12.1.11 • Testing your Rhasspy setup | 266 |

| | |
|---|------------|
| 12.2 • A Rhasspy base with satellites | 267 |
| 12.2.1 • Hardware requirements | 268 |
| 12.2.2 • Setting up the satellites | 269 |
| 12.2.3 • Setting up the base | 270 |
| 12.2.4 • Testing your base and satellites | 271 |
| 12.2.5 • Enable UDP audio streaming | 273 |
| 12.3 • Train your sentences | 274 |
| 12.3.1 • Rhasspy's template language | 276 |
| 12.3.2 • Slots | 277 |
| 12.4 • Intent handling | 278 |
| 12.4.1 • Intent handling with MQTT | 278 |
| 12.4.2 • Intent handling with AppDaemon and MQTT | 280 |
| 12.4.3 • Intent handling with WebSocket in Node-RED | 282 |
| 12.5 • Summary and further exploration | 287 |
| Chapter 13 • Remote access | 289 |
| 13.1 • Three ways for remote access | 289 |
| 13.1.1 • Port forwarding | 289 |
| 13.1.2 • A localhost tunneling solution | 293 |
| 13.1.3 • A virtual private network (VPN) | 295 |
| 13.2 • Updating your dynamic DNS with ddclient | 297 |
| 13.3 • Running WireGuard on your Raspberry Pi | 298 |
| 13.3.1 • Installing PiVPN | 299 |
| 13.3.2 • Adding a VPN client | 300 |
| 13.3.3 • Connecting with a VPN client | 302 |
| 13.3.4 • Managing your VPN clients | 304 |
| 13.4 • Summary and further exploration | 305 |
| Chapter 14 • Conclusion | 306 |
| 14.1 • A dashboard for all your services | 306 |
| 14.2 • More about home automation | 310 |
| • Appendix | 312 |
| 15.1 • Getting the name and ID of a serial device | 312 |
| 15.2 • Switching USB ports | 313 |
| 15.3 • Disabling the onboard radio chips | 313 |
| 15.3.1 • Disabling onboard Bluetooth | 314 |
| 15.3.2 • Disabling onboard Wi-Fi | 314 |

| | |
|--|-----|
| 15.4 • Disabling the on-board LEDs | 314 |
| 15.4.1 • Raspberry Pi Zero (W) | 315 |
| 15.4.2 • The big Raspberry Pi models | 315 |
| 15.4.3 • Ethernet models | 315 |
| 15.4.4 • Raspberry Pi Camera Module. | 316 |
| 15.5 • Securing insecure web services with a reverse proxy | 317 |
| 15.5.1 • Using nginx as a reverse proxy with HTTPS | 317 |
| 15.5.2 • Adding basic authentication to nginx | 321 |
| 15.6 • Bridging two MQTT brokers securely. | 323 |
| • Index | 327 |