

INDEX

Symbols

../, 218
\\.\, 218
/etc/passwd, 65, 304
/etc/shadow, 260
' OR 1=, 65, 204, 255, 268

A

Amass, 97, 125, 131–132
Amazon, 10
 Amazon Web Services, 10, 46, 110,
 280–284, 324
 API gateway, 280
API, xxv–xxvi, 25–26, 27–39
 authentication, 6–9, 19, 22, 28–32,
 42–48, 156–157, 164–165, 174,
 179–185, 197, 322
 API keys, 43–45, 57, 97–98,
 126, 133–136, 250,
 312–313
 basic authentication, 43–44,
 180, 322
 base64 authentication, 185–186
 brute force, 180–186
 classic attacks, 180–186
 HMAC, 46
 JSON Web Token. *See* JWT
 OAuth, 47–48, 324
 password reset, 57, 164, 179,
 181–183, 197
 password spraying, 170, 179,
 183–185
 verbose messaging, 54–55, 57,
 62, 64, 143, 170, 183, 202
authorization, 4, 6–7, 19, 28, 33,
42–43, 47–48, 86, 147
exploiting, 223–235
forging tokens, 187–192
hacking tips, 230–231

 token analysis, 187–190
 token capture, 189–190
data interchange formats, 39–42
gateway, 28
restaurant, xxv–xxvi
reverse engineering, 161–164
security testing, xix, 3–13, 323, 325
stateless, 31, 267–268
types, 30–38
validating, 142–152
application programming interface.
 See API
Arjun, 96, 102–103, 237, 241–242
AssetNote, 98–99, 141, 146, 326
attack surface, 3, 5, 7, 23, 97, 124–125,
287, 319, 321
authorization, 19. *See also* API:
 authorization
 to test, 4
AWS (Amazon Web Services), 10, 46,
110, 280–284, 324

B

battering ram, 82–83
Becher, Nicole, 112
BFLA, 59–61, 227–231, 242, 315, 322
 A-B-A testing, 227–228
 finding, 227–230
Bhagwat, Omkar, 313–314
Biden, Joe, xviii, 308
big-list-of-naughty-strings.txt, 204, 213
black box testing, 5–7, 102, 148–152
BOLA, 55–56, 59, 83, 103–107, 223–235,
300, 311, 317–318, 322, 325
 A-B testing, 225–226
 finding, 223–227
 side-channel, 226–227
broken function level authorization.
 See BFLA

- broken object level authorization.
 - See* BOLA
 - broken user authentication, 56–58
 - bug bounty, 5, 9, 11–12, 312–318
 - Instagram GraphQL BOLA, 317–318
 - price of good API keys, 312–313
 - private API authorization issues, 313–315
 - program, 5
 - Bug Crowd, 312
 - Burp Suite, 71, 75–84, 95–96, 103, 137–138, 142
 - BApp Store, 281
 - InQL, 298–299
 - IP Rotate, 84, 280–284
 - Comparer, 78, 205–206, 221, 317
 - Decoder, 78, 193, 195, 272, 292–293
 - Extender, 79, 83–84, 281, 298
 - with FoxyProxy, 79–80
 - intercepting traffic, 76, 79–80
 - Intruder, 78, 81–84, 105–106, 182–186, 190–192, 244–245, 273–274, 277–278
 - attack types, 82–83
 - payloads, 81–83, 106, 184–186
 - resource pool, 277–278
 - results, 245, 264, 303
 - with Kiterunner, 166–167
 - learn, 79
 - Match and Replace, 230–231, 273
 - with Postman, 95–96, 103–106
 - Repeater, 80, 142–143, 244, 298–299
 - Scanner, 79
 - scope, 78
 - Sequencer, 78, 187–189
 - site map, 78
 - target, 78
 - business logic flaw, xix, 8, 66–67, 173–174, 322
- C**
- Candelario, Ace, 312–313
 - capture the flag, 74, 115–116, 319
 - CDN (content delivery network), 269
 - cluster bomb, 82–83, 184, 191, 217–218, 260
 - content delivery network, 269
 - Common Vulnerabilities and Exposures, xviii, 320
 - completely ridiculous API, 111–112
 - Corneo, Don, 58
 - crAPI (completely ridiculous API), 111–112
 - crawling URIs, 143–145
 - create, read, update, delete, 30–32, 36, 50
 - cross-API scripting (XAS), 249, 252–253, 322, 326
 - cross-site scripting (XSS), 63, 203, 251–252, 272, 322
 - CRUD (create, read, update, delete), 30–32, 36, 50
 - CTF (capture the flag), 74, 115–116, 319
 - cURL, 90, 94, 97, 99
 - Curry, Sam, 315–317
 - CVE (Common Vulnerabilities and Exposures), xviii, 320
- D**
- Damn Vulnerable GraphQL
 - Application, 113–114, 285–292
 - databases, 23
 - nonrelational, 24–25
 - NoSQL, 24–25, 64, 203, 249
 - relational, 23–24
 - SQL, 23–24
 - data breaches, 308–312
 - Peloton, xviii, 308–309
 - T-Mobile, 311–312
 - USPS, xviii–xix, 309–311
 - data interchange formats, 39–42
 - JSON, 39–41
 - XML, 41–42
 - YAML, 42
 - DDoS (distributed denial of service), 9
 - Death Star, xxiv, 307
 - denial of service, 8–11, 59, 276
 - testing, 8, 10–11
 - developer tools (DevTools), 72–74, 138–142
 - DevSlop. *See* OWASP: DevSlop
 - directory brute force, 290
 - directory traversal, 218, 316
 - distributed denial of service, 9

Docker installation, 110–111
DoS. *See* denial of service
DVGA (Damn Vulnerable GraphQL
Application), 113–114, 285–292

E

The Economist, xxiv
evasive techniques, 267, 270
 burner accounts, 270–271
 case switching, 271, 278, 322
 encoding payloads, 272, 322
 origin header spoofing, 279
 path bypass, 278
 string terminators, 278, 322
 User-Agent, 279–280
excessive data exposure, 58, 166, 169,
 172–173, 174, 178, 309–310,
 322, 325
exposed secrets, xxvi, 321
Extensible Markup Language, 33–34,
 37–38, 41–42

F

Fair, Zack, 56
Farhi, Dolev, 113, 287
Foley, Jeff, 97
FoxyProxy, 76–77, 79, 85, 95, 163
fuzzing, 75, 83, 100–102, 201–219, 301,
 317, 325
 Burp Suite, 210–212
 bypass input sanitization, 217–218
 detecting anomalies, 204–205
 directory traversal, 218
 improper assets management,
 214–216
 metacharacters, 255, 257, 271
 payloads, 203–204
 big-list-of-naughty-strings.txt,
 204, 213
 with Postman, 207–209
 symbols, 204
 Wfuzz, 204, 212–214, 216–217
 wide and deep, 207–218

G

Gariché, Nancy, 112
Gartner, xviii, xxiv
Git, 72

GitHub, 11, 16, 29, 97, 100, 102, 114,
 125, 128, 131
Gobuster, 98, 145–146, 290
Golang, 72
Google, 11, 25, 74, 125–126, 157
 Cloud, 10, 110
 dorking, 125–126, 157
 hacking, 125–126, 157
GraphQL, xxiv, xxv, 30, 34–37, 83–84,
 113–115, 285–291, 294–298,
 308–309, 317–318, 326
 active reconnaissance, 287
 API analysis, 297
 command injection, 301–305
 cookie tampering, 292–293
 documentation, 292–293
 DVGA, 113–114, 285, 287–293,
 295–298, 301, 305
 GraphiQL, 36–37, 114, 290–298
 InQL Burp Extension, 298–299
 Introspection, 295–297, 326
 mutation, 36, 286–287, 301–302, 326
 query, 34–36, 286–287, 296–299
 requests, 35–36, 286, 294, 296–298
 response, 35–36
 reverse engineering, 290–296
 root types, 297
 subscription, 36, 286

H

HackTheBox, 115–116
Harrison, Brock, 243
HMAC (hash-based message
 authentication code), 46
HTTP (HyperText Transfer Protocol)
 methods, 17, 20–22, 30–31, 60, 88,
 157, 201, 216
 requests, 17, 75–78, 81
 responses, 18–20
 stateful/stateless, 22–23, 31, 43, 57,
 267–268
 status codes, 18–20, 170, 323
HTTP Strict Transport Security
 (HSTS), 76

I

IBM, xviii
IDOR attack, 278

- improper assets management, 65–66, 207–210, 214, 219–221
- information disclosure. *See*
 - vulnerabilities: information disclosure
- injection, 64–65, 249–250
 - cross-API scripting (XAS), 63, 249, 252–253, 322
 - cross-site scripting (XSS), 63, 203, 249, 251–252, 322
 - NoSQL, 257–259, 261–264
 - operating system command
 - injection, 259–261
 - SQL, 24–25, 64, 82, 249, 253–257, 326
 - vulnerabilities. *See* vulnerabilities: injections

J

- Janca, Tanya, 112
- JavaScript, 140, 251, 312–313
- JSON (JavaScript Object Notation), 33, 39–42
- JSON Web Token. *See* JWT
- JSON Web Token Toolkit, 194–199
- Juice Shop. *See* OWASP: Juice Shop
- JWT, 45–46, 124, 158, 179, 192–200, 325
 - abuse, 192–200
 - algorithm switch attack, 195–196
 - Crack attack, 197–200
 - JWT_Tool, 194–199
 - None attack, 195

K

- Kali Linux, 72, 323–324
- Katchum, Ash, 242
- Kimminich, Björn, 112
- Kiterunner, 75, 98–99, 146–148, 165–167, 290
- Kraushar, Mordecai, 112

L

- lack of resources and rate limiting, 59
- Li, Vickie, 5, 323

M

- mass assignment, 61–62, 237–243
- McKinnon, Connor, 113

- metacharacters, 255, 257, 271
- MFA (multifactor authentication), 181, 242–243
- Microsoft, 10–11, 24, 193
 - Azure, 10
 - Graph, 316–317
 - SQL Server, 24
- multifactor authentication, 181, 242–243
- MongoDB, 25

N

- Nikto, 99–100, 118–119, 288
- Nmap, 116–117, 138, 149–151

O

- OAS (OpenAPI Specification), 39
- OpenAPI Specification, 39
- open-source intelligence, 5, 124–125, 131, 133
- Open Web Application Security Project. *See* OWASP
- OSINT (open-source intelligence), 5, 124–125, 131, 133
- OWASP, 53–54
 - Amass, 97, 125, 131–132
 - API Security Project, 54
 - API Security Top 10, 54, 111, 113, 324
 - DevSlop, 111–112
 - Juice Shop, 112–113
 - ZAP, 96, 100, 143–145

P

- password reset, 57, 164, 179, 181–183, 197
- password spraying, 170, 179, 183–185
- payload
 - encoding, 185–186
 - types, 182, 191
 - position, 81–83
 - processing rules, 273–274
- Peloton, xviii, 308–309
- penetration testing, xxiv, 3, 5, 9–10, 323–324
- pitchfork, 82–83, 279
- Pixi, 112, 158–161, 165–169
- Postman, 84–96, 104–105, 159–167, 207–211, 219–221, 294, 324

- authorization, 86
- with Burp Suite, 95–96
- code snippets, 94, 313
- collection, 90–94, 159–165
- collection variables, 230
- headers, 86–88
- parameters, 86
- request and response panel, 87–88
- request builder, 86–89
- tests, 95
- tests panel, 94–95, 209
- variables, 86, 88–89, 91, 157–161, 207–209

Professor Hojo, 310

proxy, 76, 78–79, 85, 95–96, 105, 163–164, 166, 269, 292, 294

Python, 72

R

rate limiting, 8, 13, 28, 59, 65, 98, 157, 179, 181–183, 267, 276, 278–279, 326

rate limit testing, 8, 267, 276, 322, 326

reconnaissance, 5, 97, 124–125, 136, 138, 143

- active, 136, 138, 143
- active recon process, 136–138
- GitHub, 133–136
- passive, 124–125
- passive recon process, 125

Representational State Transfer. *See* REST

reporting, 11–12, 312, 323

REST, 30–33

- constraints, 30–31
- specifications, 38–39
- OpenAPI Specification, 39

restrictions, 9–12

reverse engineering APIs, 161–164

- GraphQL, 294–296

Rhino Security Labs, 280

robots.txt, 118–119, 139

rotating IP addresses, 280–284

S

scope

- Bug Bounty, 11–12
- Burp Suite, 78
- testing, 3–13

Shinra, Rufus, 310

Shkedy, Inon, 54, 111, 324–325

side-channel attacks, 226–227

Simple Object Access Protocol, 31, 37–38

sniper, 82–83, 211

SOAP (Simple Object Access Protocol), 31, 37–38

social engineering, 9, 43, 157

SQL injection. *See* injection: SQL

Starbucks, 315–316

Strife, Cloud, 55, 58

Swagger, 39, 90, 98, 159–160

T

testing restrictions, 9–12

testing scope. *See* scope: testing

threat actor, 4–5

threat modeling, 4–6

T-Mobile, 311–312

token forgery, 187–192

TryHackMe, 115–116

Twitter, 8–9, 17–19, 40–42, 47–50, 79–81, 131, 183

U

uniform resource locator (URL), 16–17

User-Agent, 18

US Postal Service (USPS), xviii–xix, 309–311

- Informed Visibility, xviii, 309–310

V

vulnerabilities, xviii–xix, 6–8, 11, 53–67, 83, 99–101, 155, 166, 170, 174, 201–203, 207, 214–215, 250, 301, 307–318, 326

BFLA (broken function level authorization), 59–61

- A-B-A testing, 227–228
- finding, 227–230

BOLA (broken object level authorization), 55–56

- A-B testing, 225–226
- finding, 223–227
- resource IDs, 224–225
- side channel, 226–227

broken user authentication, 56–58

- business logic, 66–67
 - finding, 173–174
 - excessive data exposure, 58
 - finding, 172–173
 - improper assets management, 65–66, 221
 - information disclosure, 8, 54–55, 62, 65, 133, 166, 169–171, 296, 322
 - verbose errors, 170–171, 202, 257, 259
 - injections, 64–65, 202
 - discovery, 250
 - cross-API scripting (XAS), 252–253
 - cross-site scripting (XSS), 251–252
 - NoSQL, 257–259, 261, 264, 326
 - operating system command, 259–260
 - SQL, 253–257
 - SQLmap, 256–257
 - lack of resources and rate
 - limiting, 59
 - testing. *See* rate limit testing
 - mass assignment, 61–62
 - automating testing, 241–242
 - blind attacks, 241
 - finding, 238–239
 - unauthorized access, 238–239
 - variables, 239–241
 - security misconfigurations, 62–64
 - encryption, 171
 - finding, 170–172
 - vulnerability reporting, 11–12, 312, 323
- W**
- WAF (web application firewall), 7, 84, 98, 218
 - Wayback Machine, 131, 157
 - web application firewall, 7, 84, 98, 218
 - Wfuzz, 100–102, 180–182, 191–204, 212–214, 216, 251, 260, 274–275
 - white box testing, 5–8, 321
- X**
- XAS (cross-API scripting), 249, 252–253, 322, 326
 - XML (Extensible Markup Language), 33–34, 37–38, 41–42
 - XSS (cross-site scripting), 63, 203, 251–252, 272, 322
- Y**
- Yalon, Erez, 54, 111
 - YAML Ain't Markup Language (YAML), 39, 42
- Z**
- ZAP. *See* OWASP: ZAP
 - zero day, xix, xxiii