SEC542 – Web App Penetration Testing and Ethical Hacking

Topics

Introduction and Information Gathering

Wy the Web? 1–7-10
DApplication Assessment Methodologies 1–11-30
Web Application Pen Tester's Toolkit 1–31-37
Interception Proxies 1–37-55
BONUS EXERCISE: Testing and Exploiting Heartbleed 1–160
EXERCISE: Configuring Interception Proxies 1–56-57
EXERCISE: Gathering Server Information 1–156-157
EXERCISE: Testing HTTPS 1–138-139
HTTP Syntax and Semantics 1–90-124
HTTPS and Testing for Weak Ciphers 1–125-137
Summary 1–158-159
Target Profiling 1–140-155
Open-Source Intelligence (OSINT) 1–58-71
Virtual Host Discovery 1–72-87
EXERCISE: Virtual Host Discovery 1–88-89

Content Discovery, Authentication, and Session Testing

Insufficient Loggin and Monitoring 2–7-11
Spidering Web Applications 2–12-23
EXERCISE: Web Spidering 2–24-25
Forced Browsing 2–26-32
EXERCISE: ZAP and fuff Forced Browse 2–33-34
Fuzzing 2–35-41
Information Leakage 2–42-48
Authentication 2–49-76
EXERCISE: Authentication 2–77-78
Username Harvesting 2–89-90
Burp Intruder 2–91-96
EXERCISE: Fuzzing with Burp Intruder 2–97-98
Session Management 2–99-114
EXERCISE: Burp Sequencer 2–115-116
Authentication and Authorization Bypass 2–117-125
Vulnerable Web Apps: Mutillidaw 2–126-130
EXERCISE: Authentication Bypass 2–131-132
Summary 2–133-134
Appendix: Shellshock 2–135-144
Exploiting Shellshock 2–145

Injection

HTTP Response Security Controls 3–7-18
Command Injection 3–19-32
EXERCISE: Command Injection 3–33-34
File Inclusion and Directory Traversal 3–35-40
EXERCISE: Local/Remote File Inclusion 3–41-42
Insecure Deserialization 3–43-61
EXERCISE: Insecure Deserialization 3–62-63
SQL Injection Primer 3–64-81
Discovering SQL 3–82-103
Exploiting SQLi 3–104-125
EXERCISE: Error-Based SQLi 3–126-127
SQLi Tools 3–128-143
sqlmap + ZAP 3–144-145
Summary 3–146

XSS, SSRF, and XXE

Document Object Model (DOM) 4–7-16
Cross-Site Scripting (XSS) Primer 4–17-31
EXERCISE: HTML Injection 4–32-33
XSS Impacts 4–34-42
BeEF 4–43-49
EXERCISE: BeEF 4–50-51
Classes of XSS 4–50-73
EXERCISE: DOM-Based XSS 4–74-75
Discovering XSS 4–76-89
XSS Tools 4–90-98
EXERCISE: XSS 4–99-100
AJAX 4–101-115
Data Attacks 4–116-121
REST and SOAP 4–122-130
Server-Side Request Forgery (SSRF) 4–131-138
EXERCISE: Server-Side Request Forgery 4–139-140
XML External Entities (XXE) 4–141-150
EXERCISE: XXE 4–151-152
Summary 4–153

CSRF, Logic Flaws, and Advanced Tools

Cross-Site Request Forgery 5–7-14
EXERCISE: CSRF 5–15-16
Logic Flaws 5–17-21
Python for Web App Pen Testers 5–22-38
EXERCISE: Python 5–39-40
WPScan and ExploitDB 5–41-43
EXERCISE: WPScan and ExploitDB 5–44-45
Burp Scanner 5–46-63
Metasploit 5–64-78
EXERCISE: Metasploit 5–79-80
Nuclei 5–81-87
EXERCISE: Nuclei/Jenkins 5–88-89
When Tools Fail 5–90-96
EXERCISE: When Tools Fail 5–97-98
Business of Penetration Testing: Preparation 5–99-112
Business of Penetration Testing: Post Assessment 5–113-123
Summary 5–124-125
BONUS EXERCISE: Bonus Challenges 5–126

Workbook

Exercise 1.1 - Configuring Interception Proxies W–2-22
Exercise 1.2 - Virtual Host Discovery W–23-37
Exercise 1.3 - Testing HTTPS W–38-43
Exercise 1.4 - Gathering Server Information W–44-54
Exercise 1.Bonus - Testing and Exploiting Heartbleed W–55-63
Exercise 2.1 - Web Spidering W–64-77
Exercise 2.2 - ZAP and fuff Forced Browse W–78-

87
- Exercise 2.3 - Authentication W–88-101
- Exercise 2.5 - Fuzzing with Burp Intruder W–117-126
- Exercise 2.6 - Burp Suquencer: Analyzing Session Tokens W–127-141
- Exercise 2.7 - Authentication Bypass W–142-165
- Exercise 2.Bonus - Exploiting Shellshock W–167-176
- Exercise 2.Bonus-2 - Snake Challenge W–177-188
- Exercise 3.1 - Command Injection W–189-206
- Exercise 3.2 - Local/Remote File Inclusion W–207-214
- Exercise 3.3 - Insecure Deserialization W–215-229
- Exercise 3.4 - Error-Based SQLi W–230-236
- Exercise 3.5 - sqlmap + ZAP W–237-250
- Exercise 4.1 - HTML Injection W–251-260
- Exercise 4.2 - BeEF W–261-275
- Exercise 4.3 - DOM-Based XSS W–276-284
- Exercise 4.4 - XSS W–285-297
- Exercise 4.5 - Servre-Side Request Forgery W–298-316
- Exercise 4.6 - XXE W–317-329
- Exercise 5.1 - CSRF W–330-346
- Exercise 5.2 - Python W–347-355
- Exercise 5.3 - WPScan and ExploitDB W–356-363
- Exercise 5.4 - Metasploit W–364-376
- Exercise 5.5 - Nuclei and Jenkins W–377-393
- Exercise 5.6 - When Tools Fail W–394-405
- Exercise 5.Bonus - Python W–406-409
- Exercise 5.Bonus2 - Exploiting Ruby on Rails W–410-422

#

Tools

- base32 W–204
- BeEF W–261
  - DOM-Based XSS W–283-284
- Burp Suite
  - auth bypass W–142-165
  - BApp - Retire.js W–51-53
  - blind injection W–201
  - Collaborator W–200-205
  - fuzzing W–118-126
  - insecure deserialization W–219-228
  - LFI/RFI W–207-212
  - request params W–177-182
  - shellshock W–169-173
  - smart decode W–372-375
  - spidering W–71-77
  - SSRF W–307-309
- crt.sh W–34-36
- curl
  - insecure deserialization W–228
  - shellshock W–173-176
  - ssrf W–305-309
  - XXE W–320-329
- diff
  - compare sitemaps W–66
- dig
  - search for record type ANY W–25
  - zone transfer W–30
- dnsdumpster W–32-33
- DNSRecon
  - dns brute W–30
  - zone transfer W–28
- fuff
  - force browse directories W–303-304
  - forced browse W–85-86
- grep
  - nmap heartbleed output W–43
- heartbleed.py W–60-61
- id
  - command injection W–192
- ifconfig
  - get VM ip W–189
- iptables
  - block icmp W–199-200
- jenkins-credentials-decryptor W–392
- Metasploit W–364
  - exploit Jenkins W–387-390
  - exploit mediawiki W–396-399
  - exploit Ruby on Rails W–410-422
- nc
  - backdoor listener W–213
  - craft HTTP request W–46-48
  - shell W–193-194
- netstat.cgi W–167-168
- nmap
  - dns-brute W–29, W–31
  - dns-zone-transfer W–28
  - enum-ciphers W–40, W–40-43
  - http-robots.txt W–50
  - NSE scripts W–50
  - script-help W–41
  - ssl-heartbleed W–55-57
  - version scan W–49
- nslookup
  - blind injection W–202
  - Burp Collaborator W–202
  - command injection W–191
- Nuclei
  - run scan W–380
  - run scan with cookie W–385
- php
  - cmd shell W–400
  - shell W–210-212
- ping
  - blind injection W–196-197, W–198-199
- python
  - generate session tokens W–134-135
  - http server W–281
  - password guessing W–406
- sqlmap

        count records in table ... W–245-47
        dump db users and password hashes ... W–249
        dump dbs ... W–311
        dump table contents ... W–313
        identify vulnerable param ... W–243
        level ... W–310
        read file ... W–248
        search col names ... W–250
        ssrf ... W–310-314
        user-agent ... W–228
    ssllabs.com ... W–39
    strings ... W–63
    tcpdump
        blind injection ... W–198-199
    tr
        -d ... W–204
    user_enum ... W–104-105
    wc ... W–30-31
    wget
        spidering ... W–65-66
    wpscan ... W–358-360
    ysoserial
        insecure deserialization ... W–219
    ZAP
        basic auth ... W–89-94
        CSRF ... W–334-336
        digest auth ... W–94-96
        forced browse ... W–81-85
        form-based auth ... W–96-98
        method interchange ... W–98-100
        spidering ... W–67-71
        SQLi error ... W–242
        user enum ... W–106-115

A

access control problems ... 1–15
Acunetix Vulnerability Scanner ... 1–35
Administrative Pages ... 1–152
AJAX ... 4–102-103
    Penetration Testing ... 4–112
    proxy ... 4–109
    Spidering ... 2–21
    XMLHttpRequest → XMLHttpRequest ... 4–104
Algorithms ... 1–131
    Assymetric ... 1–131
        DH ... 1–131
        DSA ... 1–131
        ECDH ... 1–131
        ECDSA ... 1–131
        RSA ... 1–131
    Hashing
        MD5 ... 1–131
        SHA ... 1–131
        SHA256 ... 1–131
        SHA384 ... 1–131
    Symmetric
        AES ... 1–131
        DES ... 1–131
        TDES/3DES ... 1–131
American Fuzzy Lop ... 1–9
Application Frameworks ... 1–151
Application Information Gathering
    Spidering ... 2–13
Application Infrastructure ... 1–150
Application Security ... 1–10
    Assessing ... 1–12-13
    Dynamic Application Security Testing (DAST) 1–13, 1–17-18
    Improving ... 1–12
    Interactive Application Security Testing (IAST) 1–13
    Out-of-Band Application Security Testing (OAST) ... 1–13
    Static Application Security Testing (SAST) . 1–13, 1–16
    Testing
        Full Knowledge ... 1–13
        Zero-Knowledge ... 1–13
Arachni Scanner ... 1–35
ARPANET ... 1–92
Assessments
    Automated ... 1–20
    Frameworks ... 1–22
        OWASP Web Security Testing Guide ... 1–22
        Penetration Testing Execution Standard (PTES) ... 1–22
    Full-Knowledge ... 1–13, 1–21
    Manual ... 1–20
    Methodology ... 1–22
    Testing Techniques ... 1–22
    Zero-Knowledge ... 1–13, 1–21
Assymetric ... 1–131
Attack Platform ... 1–33-34
Attack platform ... 1–32
Attack strings ... 2–36
Auditing ... 1–54
Authentication ... 2–50, 1–130
    Basic ... 2–52-53
        Illustrated ... 2–54
        Issues ... 2–55
    Bearer ... 2–72
    Bypass ... 2–119
        Parameter Tampering and Direct Page Access 2–120
        SQLi ... 2–121
    Digest ... 2–56
        Illustrated ... 2–57
        Issues ... 2–58
    Form-based ... 2–63-64
        Illustrated ... 2–65
        Issues ... 2–66
    Integrated Windows ... 2–59
        Illustrated ... 2–60
        Issues ... 2–61
    OAuth ... 2–68-71

- Openid/Oauth/SAML Authentication 2–67
- Schemes 2–50

Authentication (ATHN) 1–28
Authorization
- Attacks 2–123
- Bypass 2–122
- Role Enforcement 2–124

Authorization (ATHZ) 1–28
Automated Scanning 1–20

B

backdoors 1–15
BApp Store 1–53
Bash shell 2–137
Basic Authentication 2–52-53
- Illustrated 2–54
- Issues 2–55

bcrypt 2–87
Bearer Authentication 2–72
BeEF 4–44-46
- Issuinng Commands 4–48
- Metasploit 5–70-71
- Payloads 4–49

Best 1–20
Bing 1–65
Broswers
- Bypassing Filters 4–88

Browser 1–32
Browsers 1–36
- Authentication Schemes
  - Required 2–51
- Developer Tools 4–11

Bugtraq 1–10
BuprSuite Pro Active/Live Scan 1–35
Burp Suite
- Infiltrator 1–13
- Request Parameters 1–111

Burp Suite Professional 1–38, 1–42
- AJAX 2–21
- Authenticating 2–62
- Automated Scanning 5–47-48
  - Add to Task 5–57
  - Automatic 5–60
  - Configuration 5–53-56
  - Instantiating Scans 5–52
  - Live 5–51
  - Passive 5–49
  - Push Button 5–50
  - Results 5–58-59
- BApp Store 1–53
- Collaborator 3–28
  - Blind Exfil 3–29
- Comparer 1–52
- Cookie Jar 3–138-139
- Crawling/Auditing 1–54
- Dashboard 1–43
- Decoder 1–51, 4–59
- Discover Content 2–28
- Engagement Tools 1–55
  - Cenerage CSRF PoC 1–55
  - Content Discovery 1–55
  - Find Commment / Find Scripts 1–55
  - Search 1–55
  - Simulate Manual Testing 1–55
  - Target Analyzer 1–55
- Extender 1–53
- Grep Payloads 4–94
- Heads-Up Dispaly 1–43
- Intruder 1–48, 4–60, 2–92
  - Battering Ram 2–94
  - Cluster Bomb 2–96
  - Pitchfork 2–95
  - Reflection Tests 4–92-95
  - Sniper 2–93
- Issues
  - Retesting and Remediation Verification 5–62-63
  - Vulnerability Verification 5–61
- Proxy 1–47
- Repeater 1–49
  - Inspector 1–49
- Sequencer 1–50, 2–111-112
- Spidering 2–19
- Target
  - Scope 1–44
  - Site Map 1–45
  - Site Map - Filter 1–46

business functionality testing 1–10
Business Logic (BUSL) 1–28

C

Cache-Control 1–124
CERN 1–91
Certificate Authorities 1–87, 1–132
Certificate Transparency 1–87
CEWL
- Spidering 2–22

CGI Scripts 2–139
CHAOSNET 1–81
Checklist 1–29
Cipher Suites 1–130
Cipher suites
- Testing 1–134

Clickjacking 3–15
Client Side (CLNT) 1–28
Client-side scripting languages 3–11
Cloud
- Metadata 4–137

Cloudflare Challenge 1–163
Code review 1–13
Command
- HTTP/0.9 1–95
- HTTP/1.0 1–96
- HTTP/1.1 1–97

Command Injection 3–21-23

- Blind . . . 3–24-26
  - Detetion methods . . . 3–26-27
- Exploiting . . . 3–24
- Finding . . . 3–22-23

Command Substitution . . . 3–22

Commands
- Base 32 . . . 3–29
- Base64 Encode . . . 2–53
- Blind DNS exfil . . . 3–30-32
- DELETE request . . . 1–116
- dig (many) . . . 1–81
- dig domain lookup . . . 1–80
- DNSRecon . . . 1–83-84
- echo prevent CR . . . 2–53
- Enumerate HTTP methods . . . 1–153-154
- FuFF . . . 2–30-32
- full (AXFR) zone transfer . . . 1–75
- HEAD request . . . 1–113
- heartbleed.py . . . 1–164
- nc port scan . . . 1–142
- nmap https . . . 1–135
- OPTIONS request . . . 1–116-117
- pause using ping . . . 3–26
- Python - check if installed/version . . . 5–25
- Python - POST . . . 5–36
- Python - PSSL/TLS . . . 5–37
- Python - Putting it Together . . . 5–38
- Python - retrieve Server header . . . 5–24
- Python3 - install requests . . . 5–27
- reverse DNS . . . 1–76
- Server strings . . . 1–123-124
- SHA1 hash of session id . . . 1–104
- SQL DB Fingerprinting . . . 3–105
- SQL DB Meta Info queries . . . 3–107
- SQL Stacked Queries . . . 3–109-112
- SQLi Binary/Boolean Inference Testing . . . 3–99
- SQLi equivalent string injections . . . 3–96-98
- SQLi payload . . . 3–76
- Strip off equal signs . . . 3–29
- testssl.sh . . . 1–136
- Wget
  - robots off . . . 2–18
  - set proxy . . . 2–18
- XSS Evnet Handler . . . 4–87
- XSS Existing JS . . . 4–83
- XSS HTML img onerror . . . 4–81
- XSS Tag Attributes . . . 4–82
- xxd . . . 3–51
- XXE PoC using cURL . . . 4–144-146
- ysoserial.jar . . . 3–60

Commmon Vulnerabilities and Exposures (CVE)
- CVE-2014-6269 . . . 2–138
- CVE-2014-6271 . . . 2–137
- CVE-2014-6277 . . . 2–138
- CVE-2014-6278 . . . 2–138
- CVE-2014-6286 . . . 2–138

Common Name . . . 1–87

Common Vulnerabilities and Exposures (CVE) . . 2–48

Concurrency attacks . . . 2–92

concurrency problems . . . 1–15

Confidentiality . . . 1–130

Configuration and Deployment Management (CONF) 1–28

Configuration Files . . . 1–151

Content Management Systems(CMS) . . . 1–151

Content-Security-Policy (CSP) . . . 3–17-18
- Defined . . . 3–14

Content-Type . . . 1–109

Cookie . . . 1–122

Cookies
- Protecting . . . 3–8-10
  - HttpOnly Attribute . . . 3–11
  - SameSite Attribute . . . 3–12-13
  - Secure Attribute . . . 3–10

Coomand Injection
- Non-blind . . . 3–24

Crawling . . . 2–14

Cross Origin Resource Sharing (CORS) . . . 4–110
- Headers . . . 4–111

Cross Site Request Forgery (CSRF)
- SameSite Cookie Attribute . . . 3–12-13

Cross-Site Reqeust Forgery (CSRF)
- CSRF vs XSS . . . 5–9
- Example visualized
  - Step 4 . . . 5–13
- Tokens . . . 5–13
- Visualized
  - Step 1 . . . 5–10
  - Step 2 . . . 5–11
  - Step 3 . . . 5–12
- ZAP . . . 5–14

Cross-Site Scripting (XSS) . . . 4–18
- Bypassing Broswer Filters . . . 4–88
- Classes . . . 4–53
  - Client and Server . . . 4–73
  - DOM-Based . . . 4–71-72
  - Reflected . . . 4–54-63
  - Stored/Persistent . . . 4–64-70
- Common Injection Contexts . . . 4–80
  - Existing JS . . . 4–83
  - HTML . . . 4–81
  - Tag Attributes . . . 4–82
- cross-Site aspect . . . 4–31
- Discovery . . . 4–77
- DOM Event Handler Bypass . . . 4–86
- Filter
  - Bypass/Evasion . . . 4–85
  - Tests . . . 4–84
- Fuzzing . . . 4–77
- GET -¿ POST Flaws . . . 4–98
- HTML Event Handler Injection . . . 4–87
- Injection Points . . . 4–78
- Interception Proxies . . . 4–91
  - Burp Intruder - Reflection Tests . . . 4–92-95
- PoC Payloads . . . 4–35-37, 4–89
- Reflected POST . . . 4–97

  - Reflection Tests 4–79
  - Session Abuse 4–38-40
    - Hijacking 4–39-41
  - URL Encoding
    - Burp 4–59

Cross-Site Tracing (XST) 1–114
CRUD 1–118
Cryptographic Failures 1–126
cryptographic weaknesses 1–15
Cryptography (CRYP) 1–28

D

Data
  - Attacks 4–117
  - Formats 4–118

Database
  - Fingerprinting 3–105
  - Meta Info 3–106-107

Databases 1–151
DB Fingerprinting 3–105
Decoding 1–51
Default Web Pages 1–151
DELETE 1–105
Developer comments 2–23
Diffie-Helman 1–129
Dig 1–74
dig 1–80
  - Syntax 1–81

Digest Authentication 2–56
  - Illustrated 2–57
  - Issues 2–58

DirBuster 2–29
Direct Page Access 2–120, 2–125
Directory Broswing 2–46
Directory Browsing
  - Google 2–47

Directory brute forcing 2–27
Directory Traversal 3–38
Disabled functionality 2–23
DNS
  - Blind injection 3–27
  - Data exfil 3–29-32

DNSRecon 1–77, 1–83-84
Document Object Model (DOM) 4–8
  - Browser Objects 4–15
  - JavaScript 4–12
  - Objects
    - Document 4–15
    - Window 4–15
  - Tree 4–9-10

Documentation 1–19
Domain Name System (DNS) 1–74
  - Brute Force Scans 1–77
  - EDNS 1–74
  - PTR 1–76
  - Querying Directly 1–75
  - Recon Tools 1–78
    - dig 1–80-81
    - DNSRecon 1–83-84
    - Nmap 1–82
    - nslookup 1–79
  - Repositories 1–85-86
    - Dnsdumpster.com 1–85
    - Farsight Security's DNSDB 1–85
  - Reverse 1–75
  - Zone Transfer 1–74

Dorking 1–67
Dorks 1–66
Drupal 5–74
  - Drupalgeddon 5–75-78
  - Got hacked? 5–76
  - Metasploit 5–78

DuckDuckGo 1–65
Dynamic Application Security Testing (DAST) .. 1–13, 1–17-18, 1–35
  - Scanners 1–35
  - Tools 1–17-18

Dynamic web application security scanners 1–32

E

Easter eggs 1–15
EDNS 1–74
Elasticsearch 4–138
Electromagnetic Interference (EMI) 2–84
embedded 3–15
Encoding 1–51
Enrors
  - Information Leakage 2–44

Enterprise Resource Management (ERM) 5–116
Entropy 2–112
Error Handling (ERRH) 1–28
Exploit Database (EDB) 1–10

F

FFuF 2–30-32
  - Custom Filters 2–32
  - HTTP request 2–31
  - Virtual Host Discovery 2–31-32

filter bubble 1–64
Fingerprinting 1–141
Firefox
  - URL Encoding 4–62

flawed business logic 1–15
Forced Browsing 2–27
  - BurpSuite 2–28
  - FFuF 2–30-32
  - file extensions 2–27
  - ZAP 2–29

Form-based Authentication 2–63-64
  - Illustrated 2–65
  - Issues 2–66

Fortify WebInspect 1–35

- forward proxy 1–38
- Full Disclosure 1–10
- Full-Knowledge 1–13, 1–21
- FuzzDB 2–40
- Fuzzing 2–36
  - Examples 2–37
  - Reviewing the Responses 2–38
  - XSS PoC Payloads 4–89
  - ZAP 2–41

G

- Gadget classes 3–59
- Generalized Markup Language (GML) 1–91
- GET 1–106-108
  - URI Query Parameters 1–108
- Globally Unique Identifiers (GUID) 2–103
- Google 1–65-66
  - Directory Browsing 2–47
- Google Dorks 1–66
- Google Hacking Database (GHDB) 1–66-67
  - GHDB-ID 4804 1–67
- Governance, Risk, and Compliance (GRC) 5–116
- Grep 1–48

H

- Hashing 1–51, 1–131
- HCL AppScan 1–35
- HEAD 1–113
- Heartbleed 1–161
  - Effect 1–162
  - Exploit Output 1–164
- Hidden links 2–23
- High-risk functions 2–23
- HMAC
  - JWT Signature Cracking 2–76
- HPACK 1–98-99
- HTML
  - Injection 4–20-21
- HTTP Strict Transport Security (HSTS) 3–16
  - Defined 3–14
- HTTPOnly 4–42
- HttpOnly 3–11
  - Bypass 1–114
- HTTPS
  - Authentication 1–130
  - Cipher Suites 1–130
  - Confidentiality 1–130
  - Integrity 1–130
  - Public Keys/Certificates and Certificate Authorities 1–132
  - Testing
    - nmap 1–135
    - Version and Cipher Suites 1–134
  - Transmission Security 1–127
- HTTPS Certificates 1–87
- Hypertext Transfer Protocol (HTTP)
  - 0.9 1–95
  - 1.0 1–95-96
  - 1.1 1–97
  - 3 QUIC 1–100
  - 2 1–98-99
  - History 1–9, 1–91
    - 0.9 1–9
  - QUIC 1–93
  - Request
    - Example 1–102
    - Highlights 1–123
    - Methods/Verbs 1–105-116, 1–122
    - Referer 1–121
    - User-Agent 1–120
  - Response
    - Cache-Control 1–123-124
    - Example 1–103-104
    - Server 1–123
    - Set-Cookie 1–123
    - Status Code 1–123
  - Security Headers 3–14-17
  - Semantics 1–93, 1–101
  - Syntax 1–93
  - Testing
    - Methods 1–153
  - URI Query Parameters 1–108

I

- idempotent 1–105
- Identifying Web Server Components 1–151
- Identity Management (IDNT) 1–28
- Identity Provider 2–67
- iframe 3–15
- Incident Response 2–11
- Information Gathering 1–59
- Information Gathering (INFO) 1–28
- Information Leakage 2–44
  - CVE search 2–48
  - Flaws 2–48
  - Types 2–45
- Insecure Deserialization 3–44-47
  - Exploitation 3–48
  - Java Example 3–49-57
  - RCE 3–58-59
- Insecure Direct Object Reference (IDOR) 2–125
- Insecure Directo Object Reference (IDOR)
  - AuthZ attacks 2–123
- Integrated Windows Authentication 2–59
  - Illustrated 2–60
  - Issues 2–61
- Integrity 1–130
- Integrity hashes 2–86
- Interactive Application Security Testing (IAST) 1–13
- Interception Proxies 1–38, 4–91
  - Configuring 1–57
  - Zed Attack Proxy (ZAP) 1–39

- Interception proxies 1–32
- Internet 1–91-92
- Internet Explorer (IE) 4–26
- Invicti (formerly Netsparker) 1–35

J

- JavaScript 4–13
  - Browser Objects 4–15
  - DOM Manipulation 4–12
  - Frameworks 4–113-114
    - AngularJS 4–113
    - Bootstrap 4–113
    - jQuery 4–113
    - ReactJS 4–113
  - Injection 4–22
  - Objects 4–14
    - Document 4–15
    - Window 4–15
  - Server-side 4–115
  - Web Page Interaction 4–16
- JavaScript Object Notation (JSON) 4–118, 119
  - Advantages of XML 4–118
  - Attacks 4–120-121
- JSON Web Token (JWT) 2–72-73
  - Issues 2–76
  - Structure 2–74-75

K

- Kali 1–33-34
  - metapackages 1–34
  - Update Command 1–34
- Kerberos 2–59, 1–92

L

- Link Depth 2–14
- LinkedIn Open Networkers (LIONs) 1–68
- Local File Inclusion (LFI) 3–37
  - Demonstrating Impact 3–39
  - Directory Traversal 3–38
- Logged Events and Sources 2–9
- Logging and Monitoring 2–8
  - Centralized Logs 2–10
  - Sources 2–9
  - Timestamping 2–9
- logic bombs 1–15
- Logic Flaws 2–16, 5–18
  - Manual Discovery 5–20
  - Workflow Tampering 5–19
  - WSTG Guidance 5–21

M

- Mailing Lists 1–10
  - Bugtraq 1–10
  - Full Disclosure 1–10
- Man-in-the-middle (MITM) 1–38
- Manual Assessments 1–20
- Manual Inspection and Reviews 1–19
- MD5 2–87
- MediaWiki 5–93
- META tags 2–15
- Metadata 4–137
- Metasploit 5–65
  - Auxiliary modules 5–66
  - BeEF 5–70-71
  - db_import 5–68
  - Drupal 5–78
  - Integration 5–69
  - Known Vulnerabilities 5–73
  - mediawiki_thumb 5–93, 5–96
  - Seeding Database 5–67
  - Spidering and Crawling 5–67
  - sqlmatp 5–72
  - Web Testing 5–66
- Method Interchange 1–112
- Method Tampering 1–112
- Methodology 1–22
- Micah Hoffman's OSINT MindMap 1–61
- Michal Zalewski 1–9
- MITRE 2–48
- Morris Worm 1–92
- MS SQL
  - Databases/Tables/Columns enum 3–107
- Multi-Factor Authentication (MFA)
  - Bypass 3–8
- Multiplexing 1–98
- Mutillidae 2–127
  - Hints 2–129
  - Reset DB 2–130
  - Security Levels 2–128
- MySQL
  - Databases/Tables/Columns enum 3–107
- mySQL 1–150

N

- Netcraft Sitereport 1–149
- Network Vulnerability Scanner 1–35
- Nikto 1–35
- NIST
  - 800-30
    - Appendix D 1–14
- NIST National Vulnerability Database 1–10
- Nmap
  - DNS NSE Scripts 1–82
- nmap 1–144-145
- Node.js 4–115
- nslookup 1–79
- NTML 2–59
- NTP 2–10
- Nuclei Vulnerability Scanner 5–82-83

  - Command options 5–87
  - Interactsh 5–86
  - Nuclei Templates 5–84-85
  - Running Nuclei 5–86-87

O

- OAuth 2–68
  - Flow 2–69-71
- Offensive Security 1–10
- One-Time Password (OTP)
  - Bypass 3–8
- Open-Source Intelligence (OSINT) 1–60
  - For Web Application Testers 1–62
  - Micah Hoffman's OSINT MindMap 1–61
  - Search Engines 1–64
    - Dorking 1–67
    - Google Dorks 1–66
    - Key Search Operators 1–65
  - Social Media 1–68
    - LinkedIn 1–68
  - SpiderFoot 1–71
  - Suites 1–70
- OpenID/OAuth/SAML Authentication 2–67
- OpenSSL 1–161
- Operational Controls 2–8
- Operational Security (OPSEC) 1–71
- OPTIONS 1–116
- Oracle 3–115
  - Databases/Tables/Columns enum 3–107
- Origin of Trust 4–23
- Other software 1–151
- Out-of-Band Application Secuirty Testing 1–13
- OWASP 1–24
  - Top 10 1–25-26
    - A08:2021-Software and Data Integrity Faulures 3–44
    - A09:2021-Security Loggin and Monitoring Failures 2–8
    - Cryptographic Failures 1–126
  - Web Application Penetration Testing Methodology 1–27
  - Web Security Testing Guide (WSTG) . 1–19, 1–22, 1–29
    - Information Gathering 1–59
    - Testing Categories 1–27
    - Web Application Penetration Testing Methodology 1–27

P

- p0f 1–9
- Paros Proxy 1–39
- Password hashes 2–86
- PATCH 1–118
- PBKDF2 2–87
- Penetration Testing 1–17
  - Communication Plans 5–110-112
  - Debrief Presentation 5–123
  - Post Assessment Results 5–114
  - Pre-Engagement Specifics 5–105-109
    - Backups 5–109
    - Filtering 5–108
    - Permission 5–106
    - Rules of Engagement 5–107
    - Scope 5–107
    - Source Details 5–108
    - Test Accounts 5–108
  - Preparation 5–100
  - Reports 5–115
    - Alternative Result Formats 5–121
    - Appendices 5–120
    - Automation 5–122
    - Executive Summary 5–116-117
    - Findings 5–119
    - Test Parameters 5–118
  - Skills and Practice 5–101-102
  - Toolkit 5–103-104
- Penetration Testing Execution Standard (PTES) . 1–22
- PHP
  - Expect module 4–148
- PHP Data Object (PDO) 5–77
- Ping 3–26
- Pipelining 1–98
- Port
  - 3306/tcp 1–150
- Port Scanning 1–142, 1–151
  - Tools
    - Netcraft 1–143
    - Nmap 1–143
    - Shodan 1–143
    - Zenmap 1–143
- POST 1–109
  - Parameters 1–110
- Post-exploitation 1–22
- Postman 4–128
- Private key 1–132
- Privilege Escalation 2–123
- Profiling 1–141
- Property-Oriented Programming 3–59
- Pseudo-random number generators (PRNG) 2–103
- Public key 1–132
- Push Button Pen Testing 1–18
- Push Promise 1–98
- Python 5–23
  - 2 vs. 3 5–26
  - Availability 5–25
  - Basics 5–29
  - Data Types and if/elif/else Syntax 5–30
  - for Pentesters 5–24
  - Lists and Dictionaries 5–32
  - Loops 5–31
  - use 3, right? 5–27
    - Example 5–27
  - Web Libraries 5–33

SEC542 – Web App Penetration Testing and Ethical Hacking

Requests 5–31-35
why writre scripts? 5–28

Q

quality of protection (qop) 2–56
Qualys SSL Labs 1–137
Qualys WAS 1–35
Quick UDP Internet Connection (QUIC) 1–100

R

Rapid7 AppSpider 1–35
Realm 2–52
Recon and Enum 1–141
Reconaissance 1–22
Referer 1–121
Relational Databases 3–67
Remote File Inclusion (RFI) 3–40
Reporting 1–22
REST 4–123
RESTful API 4–125
Return Oriented Programming (ROP) 3–59
Reverse DNS 1–76
RFC
2069 Digest OG 2–56
2617 Digest Auth 2–56
6749 OAuth 2–68
6750 Bearer AuthN 2–72
7231book 1 114
1918 1–92
1945 1–94-95
2246 1–127
2616 . . . . . 1–97, 1–114
2617 2–52
3986 . . . . . 1–106, 1–108
5246 1–127
5789 1–118
6265 1–114
6520 1–161
7231 1–97
7232 1–97
7233 1–97
7234 1–97
7235 . . . . . 1–97, 1–114
7540 1–98
7541 1–99
8446 1–127
9114 1–100
Robot Exclusion Protocol 2–15
Robots.txt 1–155
robots.txt 2–15
Role Enforcement 2–124
RSA 1–129
Runtime Application Self-Protection (RASP) 1–13

S

Same Origin Policy (SOP)
Evading 4–109
XMLHttpRequest 4–108
Same-Origin Policy (SOP) 4–23
Basics 4–24
Externally Sourced Scripts 4–28-29
Not Bypassing 4–30
Requirements 4–25
Test Case
Cookies 4–27
DOM 4–26
SameSite 3–12
Scanning 1–22
Script Injection 4–22
SecLists 2–40
Secure Attribute 3–10
Security Information and Event Management (SIEM) 2–10
Security Posture 1–14
Semantics 1–93
Server
Configuration flaw 2–8
Server-Side Request Forgery (SSRF) 4–132-134
Impact 4–136-138
Cloud 4–136
Session Analysis 1–50
Session Management (SESS) 1–28
Sessions
Abuse 4–38-40
Hijacking 4–39-41
Attacks 2–110
Fixation 2–113
Logout and Timeout 2–114
Predictability 2–111-112
Confidential 2–107
Expire 2–106
Identifiers 2–101
Management 2–100
Testing 2–109
Management Principles 2–102
Predicatbility 2–104
Tamper Resistant 2–105
Theft 2–108
Unpredictable 2–103
Set-Cookie 1–122-123, 1–124
Shellshock 2–137-139
fuzzing 2–94
Payloads 2–140
Execution and Impact 2–143
Injection Explained 2–141-142
Visualized 2–144
Shodan 1–146-148
for Pentesters 1–147
Port Scanner 1–148
Side-Channel Attacks 2–84
Practical 2–85
Timing 2–86
Practical 2–88

SEC542 – Web App Penetration Testing and Ethical Hacking

- Single Page Applications (SPA) → JavaScript Frameworks . . . 4–113-114
- Single Sign On (SSO)
  - Integrated Windows Authentication . . . 2–59
- Slow hashing . . . 2–87
- SOAP . . . 4–123
  - Web Services . . . 4–126-127
- SoapUI . . . 4–129
- SOCMINT . . . 1–68
- Software Development Lifecycle (SDLC) . . . 1–17, 1–19
- Software Engineering Institute (SEI) . . . 1–14
- Source Code Review . . . 1–15
  - Advantages . . . 1–15
  - Disadvantages . . . 1–15
- SpiderFoot . . . 1–71
- Spidering . . . 2–14
  - AJAX . . . 2–21
  - Analyzing Results . . . 2–23
  - BurpSuite . . . 2–19
  - CEWL . . . 2–22
  - Command-Line . . . 2–18
  - Manual and Automated . . . 2–17
  - Manual vs. Automated . . . 2–16
  - ZAP . . . 2–20
- SQL
  - Data Types . . . 3–70
  - Key Verbs . . . 3–68
  - Query Modifiers . . . 3–69
  - Special Characters . . . 3–71
- SQL Injection (SQLi)
  - Auth Bypass . . . 2–121
  - Balancing Act . . . 3–78
    - Colmun Numbers . . . 3–80
    - DataType . . . 3–81
    - Quotes . . . 3–79
  - Blind . . . 3–87-100
    - Custom Error Messages . . . 3–92-94
    - Database Error Messages . . . 3–89-91
    - Increasing Blindness . . . 3–100
    - Out-of-Band . . . 3–102-103
    - Timing Inferences . . . 3–101
  - Blind Data Exfil . . . 3–120-122
  - Cheat Sheets . . . 3–125
  - Classes . . . 3–85
  - Data Exfil . . . 3–119-122
  - DB Fingerprinting . . . 3–105
  - DB Meta Info . . . 3–106-107
  - Determine Data Types . . . 3–118
  - Discovering . . . 3–83
  - DUAL . . . 3–115
  - Examples . . . 3–72-77
  - FROMless SELECT . . . 3–115
  - In-Band/Inline . . . 3–86
    - Exploiting . . . 3–108
    - Stacked Queries . . . 3–109-112
  - Input Locations . . . 3–84
  - Intro . . . 3–65
  - NULL . . . 3–116
  - ORDER BY . . . 3–117
  - Origin . . . 3–66
  - Payloads . . . 3–76
    - Binary/Boolean Inference Testing . . . 3–99
    - equivalent String Injections . . . 3–96-98
  - Potential Attacks . . . 3–123
  - Tools . . . 3–129
    - sqlmap . . . 3–130
  - UNION . . . 3–113-114
  - UNION+NULL . . . 3–117
  - Without Error Messages . . . 3–95
  - Write file -¿ Shell . . . 3–124
- sqlmap
  - Data exfil . . . 3–120
  - Metasploit . . . 5–72
- SSL . . . 1–127
- Static Application Security Testing (SAST) 1–13, 1–16
- Status Codes . . . 1–123
- Symmetric . . . 1–131
- Syntax . . . 1–93

T

- Target
  - Profiling . . . 1–141
- Tautology . . . 3–76
- Testing Techniques . . . 1–22
- testssl.sh . . . 1–136
- The Tangled Web . . . 1–9
- Threat Modeling . . . 1–13
  - Advantages . . . 1–14
  - Disadvantages . . . 1–14
  - NIST 800-30 . . . 1–14
- Tim Berners-Lee . . . 1–9, 1–91
- time bombs . . . 1–15
- Timing Attacks . . . 2–86
  - Practical . . . 2–88
  - Slow Hashing . . . 2–87
- Timing-based attack . . . 2–38
- TLS . . . 1–127
  - Handshake . . . 1–128-129
- Token Authentiction . . . 2–72
- Tookkit . . . 1–32
- Tools
  - BeEEF . . . 4–44-46
    - Issuing Commands . . . 4–48
    - Payloads . . . 4–49
  - heartbleed.py . . . 1–164
  - Netcraft Sitereport . . . 1–149
  - Next level . . . 5–92
  - nmap . . . 1–144-145
  - Shodan . . . 1–146-147
  - sqlmap . . . 3–130
    - Auth/Sessions/Proxies . . . 3–135
    - Beyond DB Data Exfil . . . 3–142
    - DB Data Exfil . . . 3–141
    - DB Enum . . . 3–140
    - HTTP Headers . . . 3–139

Initial Targeting 3–134
Post Exploitition & Priv Esc 3–143
Proxies and Active Sessions 3–136
Riding ZAP/Burp Sessions 3–137-138
When they fail 5–91
Research the flaw 5–94
WPScan 5–42-43
Total cost of ownership (TCO) 2–83
TRACE 1–114
Transmission Security 1–127
Trojans 1–15

U

Unlinked content 2–27
URI Query Parameters 1–108
User-Agent 1–120
Username Enumeration 2–81-82
Results to look for 2–83
Side-Channel Attacks 2–84
Username enumeration 2–38
Username Harvesting 2–81-82
Results to look for 2–83
Side-Channel Attacks 2–84
UTC 2–10

V

Verb Tampering 1–112
Version Detection 1–151
Virtual Host Discovery 1–73
DNS 1–74-75
HTTPS Certificate 1–87
Vulnerability assessment 1–22
vulnerability mailing lists 1–10
Vulnerability Scans 1–151

W

Web Application PenTester's Toolkit 1–32
Web Applications 1–8
web client 1–120
Web Security Testing Guide (WSTG)
Information Gathering 2–13
Logic Flaws 5–21
Review Webserver Metafiles for Information Leakage 1–155
Test HTTP Methods 1–153-154
Testing for Weak Transport Layer Security . 1–132
WSTG-ATHN-04: Testing for Bypassing Authentication Schema 2–118
WSTG-CLNT-03:Test for HTML Injection .. 4–19
WSTG-CONF-04: Review Unreferenced Files for Sensitive Information 2–43
WSTG-IDNT-04: Testing for Account Enumeration 2–80
WSTG-INPV-11:Testing for Code Injection: LFI/RFI 3–36
WSTG-IVPV-12:Testing for Command Injection 3–20
WSTG-SESS-05:Testing for Cross-Site Request Forgery (CSRF) 5–8
Web Servers 1–151
Web Services 4–123-124
Pentesting Methodology 4–130
Wget 2–18
Whitehat Sentinel 1–35
Wordlists 2–36
WordPress 5–42-43
World Wide Web 1–91
World Wide Web (W3) 1–94
WPScan 5–42-43
WWW-Authenticate 2–51

X

X-Frame-Options 3–15
Defined 3–14
XML 4–118
XML External Entity (XEE) 4–142-143
Blind 4–149
Example 4–150
Example
Access URL 4–147
Display Local File 4–146
RCE via PHP 4–148
PoC 4–144-145
SSRF 4–135
XML Schema Definition (XSD) 4–145
XMLHttpRequest 4–104
Features and Limitations 4–108
Properties and Methods 4–105-106

Y

ysoserial 3–60

Z

ZAP Active Scan 1–35
Zed Attack Proxy (ZAP) 1–38, 39
AJAX 2–21
Attack Menu 1–41
Cookie Jar 3–138-139
CSRF Test Form 5–14
Discover Content 2–29
Encoding URLs 4–61
Functions
Active Scan 1–41
Forced browsing 1–41
Fuzzing 1–41
Spider and AJAX spider 1–41
Fuzzer 2–41
Interface 1–40
Spidering 2–20
Zero-Knowledge 1–13, 1–21
Zone Transfer 1–74-75