Take Aways

Overview of the web app penetration testing process
Web proxy tool
Reporting
Gaps in the process
What is it?

- Penetration testing vs vulnerability assessment
- Finding security issues, exploiting them, and reporting on it
FINDING VULNERABILITIES BEFORE THE BAD GUYS DO
UNDERSTANDING THE APPLICATION SECURITY POSTURE
LEGAL REQUIREMENTS (E.G PCI COMPLIANCE)

Why is it needed?
Scoping the application

- Requirements for testing
  - Effort days
  - Software/hardware requirements
  - Whitelisting
  - Testing window
  - Special requests
  - Cost
Our Methodology

1. Information gathering
2. Developing test cases
3. Vulnerability discovery & exploitation
4. Risk analysis
5. Providing support
6. Reporting
Methodology 2 – Information Gathering

- Your browser and dev tools are your best friend
- Unauthenticated vulnerabilities and exposures are the most critical
- Depending on the timeline, proceed in order of attacks that are most likely to succeed
- Try non-intrusive methods such as searching DNS records, as well as traceroute and other enumeration

*** Stakeholders need to be notified about public exposures and unauthenticated vulnerabilities right away! ***
Case study

A WordPress site running version 4.7.0 was vulnerable to Content Injection leading to an embarrassing and potentially reputation impacting message from a script kiddie.
Acting on Information Gathered

**Application walkthrough**
Discover the app’s functionality by investigating using your browser first
See how much can be found without authentication.
Look for common URLs, directories, and error pages

**Fingerprinting**
What JS framework are they using?
Sometimes session cookie names give away the underlying platform:
"JSESSIONID", "ASP.NetSessionID"

**Analyze**
Maybe you have some experience writing code in these languages
Think about how you would implement this functionality, assumptions made, corners cut, etc
Challenge what the developer’s assumptions in your testing
Developing Test Cases

Breaking components of the application by issues:
• Authentication and authorization issues
• Session management
• Data validation
• Misconfigurations
• Network Level issues

Developing Business logic test cases:
• Jumping user flows
• Testing authorization controls
Vulnerability Discovery & Exploitation

- Carrying out the test cases
- Observing application behavior
- Improvising as the test proceeds
- Google everything
Risk Analysis

Impact of a successful attack

- How much damage can it cause
- Taking business into context

Likelihood of a successful attack

- Vulnerability discovery
- Payload creation difficulty
- Any mitigating controls in place
Reporting

- Security issue description
- Evidence
- Impact/Likelihood of an attack
- Recommendations
- Presentation
- Support
Our Favorite Tool

- Burp Suite Pro:
  - Proxy HTTP traffic
  - Allows modification of URL parameters and HTTP request body
  - Useful for business logic testing
  - Easy searching of information sent or received
Gaps in the process

- Assessments are timeboxed
- Limited to the tester's technical abilities
- Test environment misrepresentation
- Narrow scopes
- Attack surface limitations
Q&A

Questions?