



Introduction to XSS

Pen and the Art of Cross-Site Scripting

Jakob Pennington
@JakobRPenny



Overview

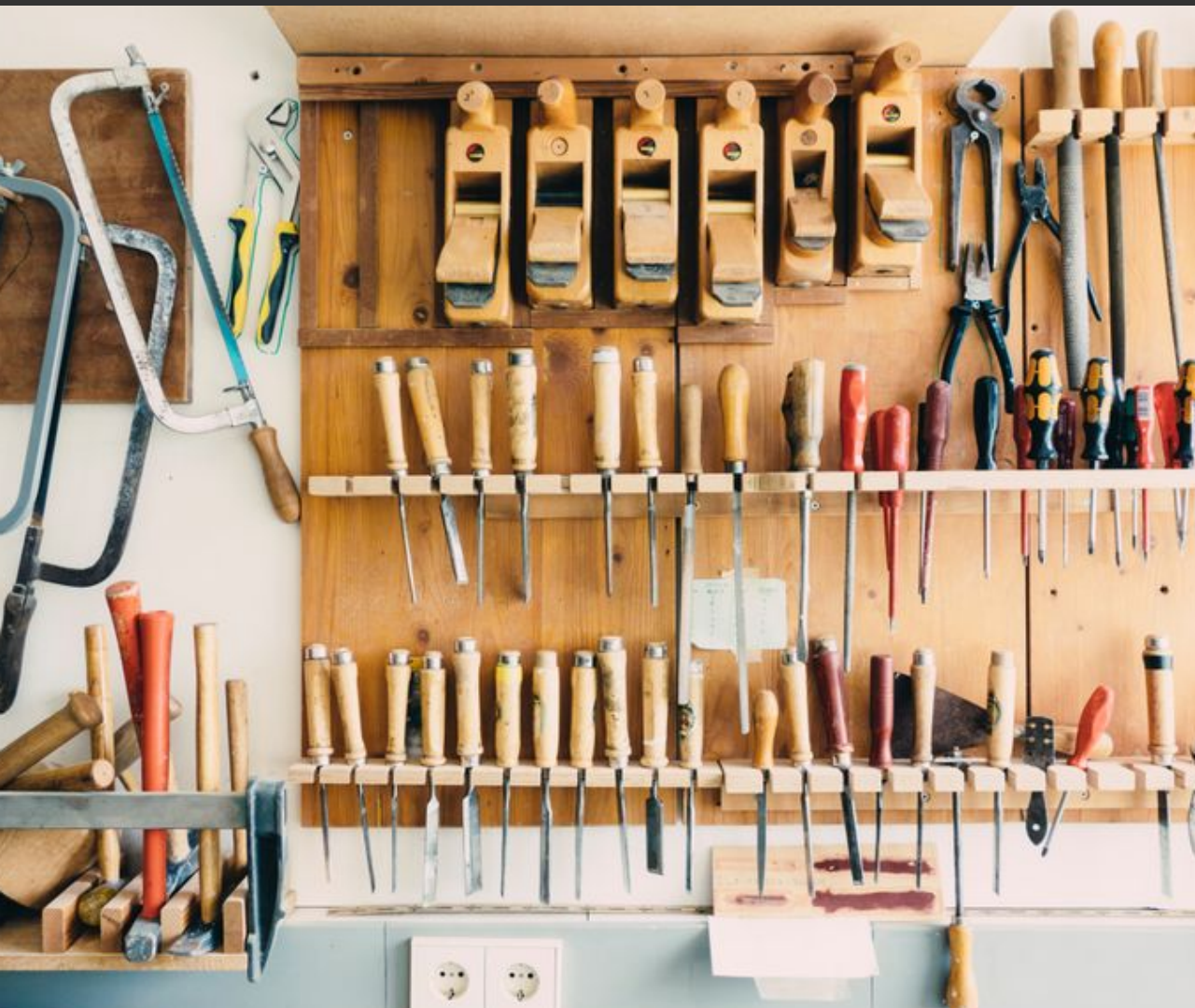
- 1 Perspective
- 2 Fundamentals
- 3 Vulnerability
- 4 Vectors
- 5 Exploit
- 6 Advanced
- 7 Mitigation
- 8 Parting

Perspective



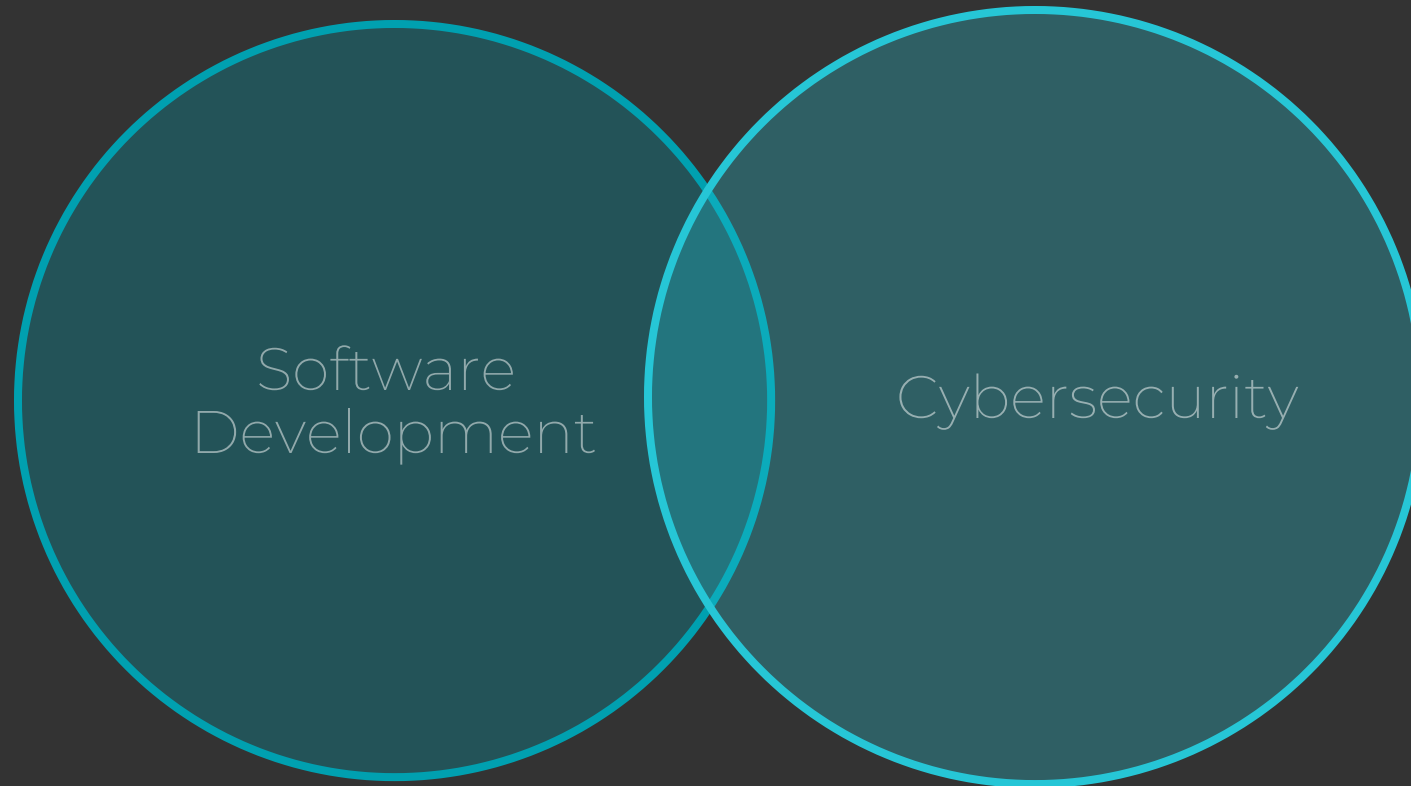
Things I Like to Do

Build Stuff + Break Stuff



Where My Interests Lie

Right in that middle bit



Building Security Tools
Secure Software Development

Fundamentals

Anatomy of a Webpage



HTML
Content

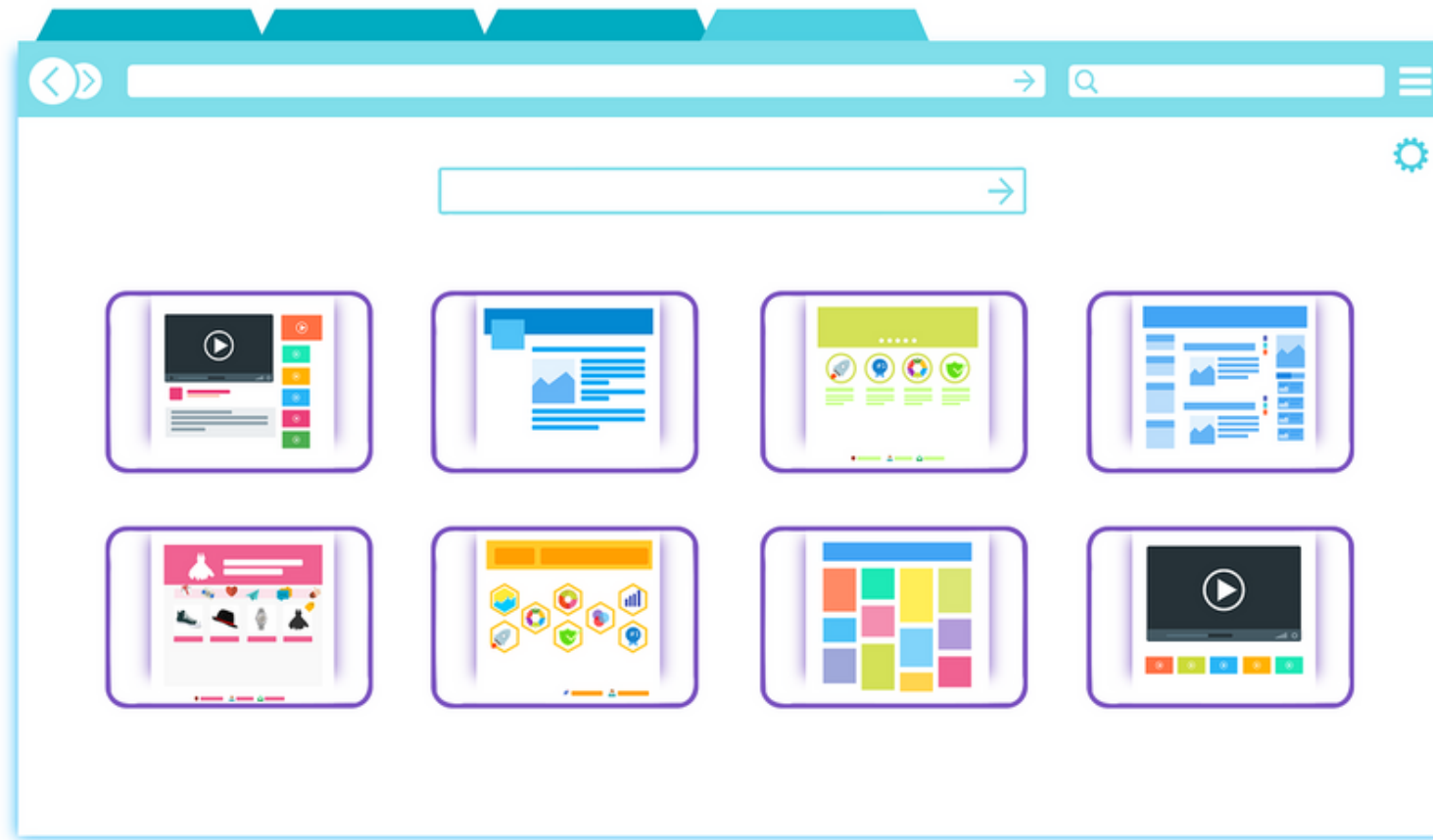


JS
Behaviour



CSS
Styling

Web Browser





Apache Cordova



Electron



React Native



Progressive Web Application

Vulnerability

OWASP Top 10

- 1 Injection
- 2 Broken Authentication
- 3 Sensitive Data Exposure
- 4 XML External Entities (XXE)
- 5 Broken Access Control
- 6 Security Misconfiguration
- 7 **Cross-Site Scripting**
- 8 Insecure Deserialisation
- 9 Using Components with Known Vulnerabilities
- 10 Insufficient Logging & Monitoring

Static Web Pages

Dynamic Web Pages

Conditions for XSS

- ① Data enters web application from untrusted source
- ② Data loaded into dynamic page without being validated

Types of XSS

Reflected



Stored



Vectors

Reflected XSS Vectors

- ① Crafted URL
- ② Malicious third-party webpage
- ③ Reflected form data

Demo

REFLECTED XSS

Stored XSS Vectors

1 User input saved to database

- Form data
- Messages / forum posts

2 Logs

Demo

STORED XSS

Demo

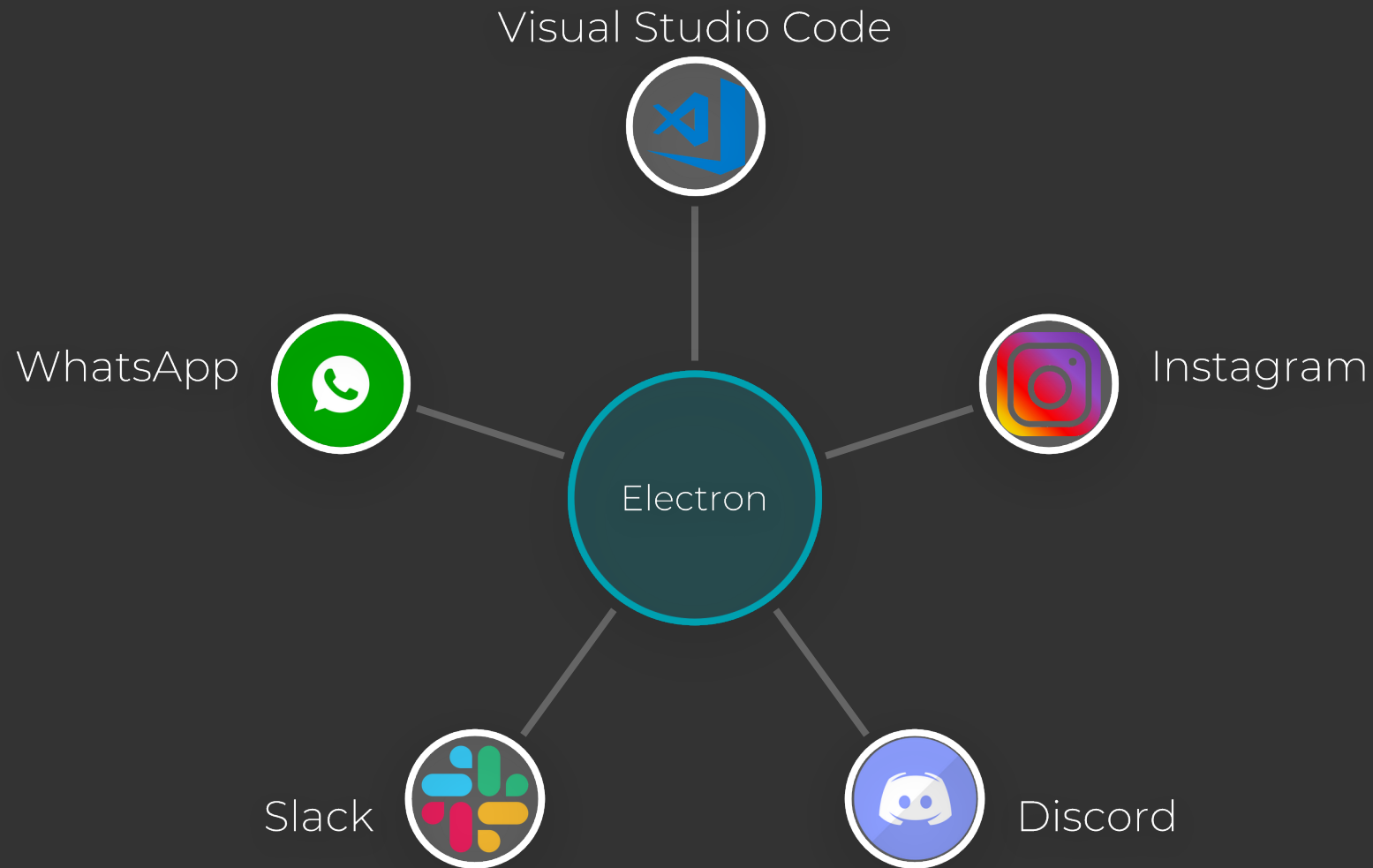
MARKDOWN

Exploits

Demo

Advanced

Electron is Everywhere



Electron RCE Demo

Advanced Attacks

- **Authenticated Requests**
Send requests to vulnerable site as victim
- **Web Proxy**
Use a webpage as another user
- **Service Workers**
Persistent JS execution after browser tab is closed
- **Internal Network Scanning**
Pingsweep and port scan internal resources
- **RCE on Internal Network**
Pivot from victim to internal network

Mitigation

XSS Mitigation Techniques (code)

- ① Encode / sanitise on save to database
- ② Encode / sanitise on inserting into web page
- ③ Parse JSON / XML safely
- ④ Use a trusted sanitisation library
- ⑤ Use a modern JS Framework

XSS Mitigation Techniques (browser)

- ① HttpOnly (cookie flag)
- ② Content Security Policy (header)
- ③ X-XSS-Protection (header)

Parting Notes

What Hasn't Been Covered

- ① Finding XSS Vulnerabilities
- ② Crafting Exploits
- ③ Bypassing WAFs
- ④ Browser Security

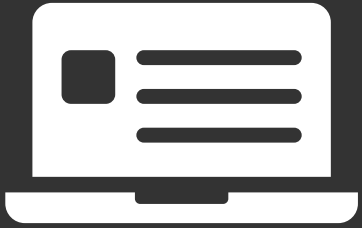
Want To Lean More?

- ① Hack the Box
- ② Pentester Labs
- ③ Cybrary
- ④ OSCP
- ⑤ Twitter

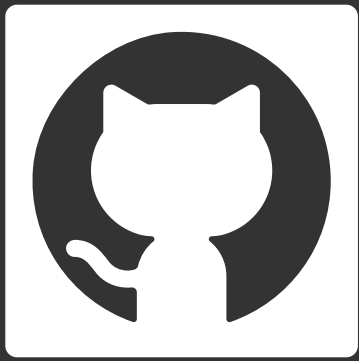
Do This Professionally

- ① Penetration Testing
- ② Bug Bounty Programs
- ③ Security Research

Thanks!



vulnerable.pennington.io



github.com/JakobRPennington/Vulnerable



meetup.com/en-AU/SecTalks-Adelaide/



@TeamHFOH
@TaptuIT

taptu @



Jakob Pennington

Application Security Specialist
@TaptuIT

 jakob.pennington@taptu.com.au

 [linkedin.com/in/jakobpennington/](https://www.linkedin.com/in/jakobpennington/)

 [@JakobRPenny](https://twitter.com/JakobRPenny)