Testing in the age of Hackathons

- SIGIST 11TH JULY 2017, DON SHARP AND SUNAINA DHANJAL

We will talk about …

- ❑Get your Hack on
- ❑Racing and Wagering WA
- ❑What's a Hackathon?
- ❑Why you would have one
- ❑What they look like
- ❑Testing in Hackathons

Get Your Hack on !

Racing and Wagering WA

Testing at RWWA - by the Numbers

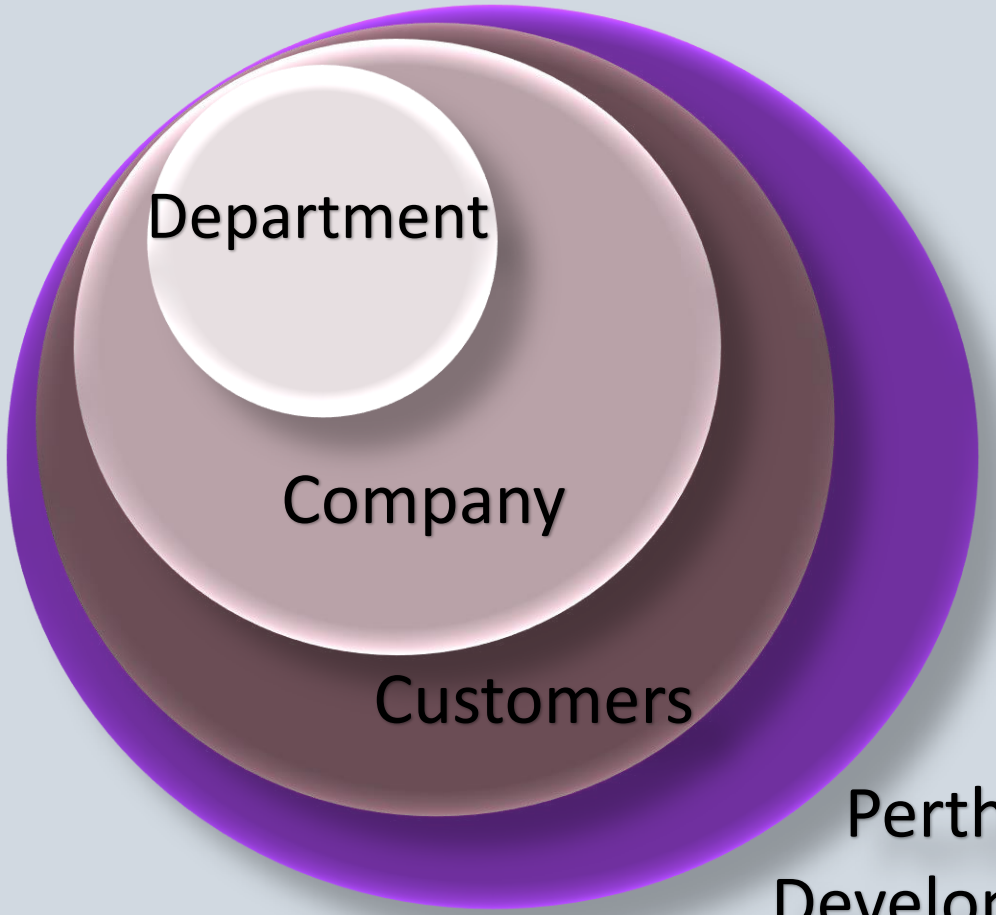
- ❑Testers: 20-30
- ❑Agile Teams: 8
- ❑Channels: 4
- ❑Systems supported: 110+
- ❑Turnover: $2.2B
- ❑Operations: 24x7

Service Level: 99.96% uptime

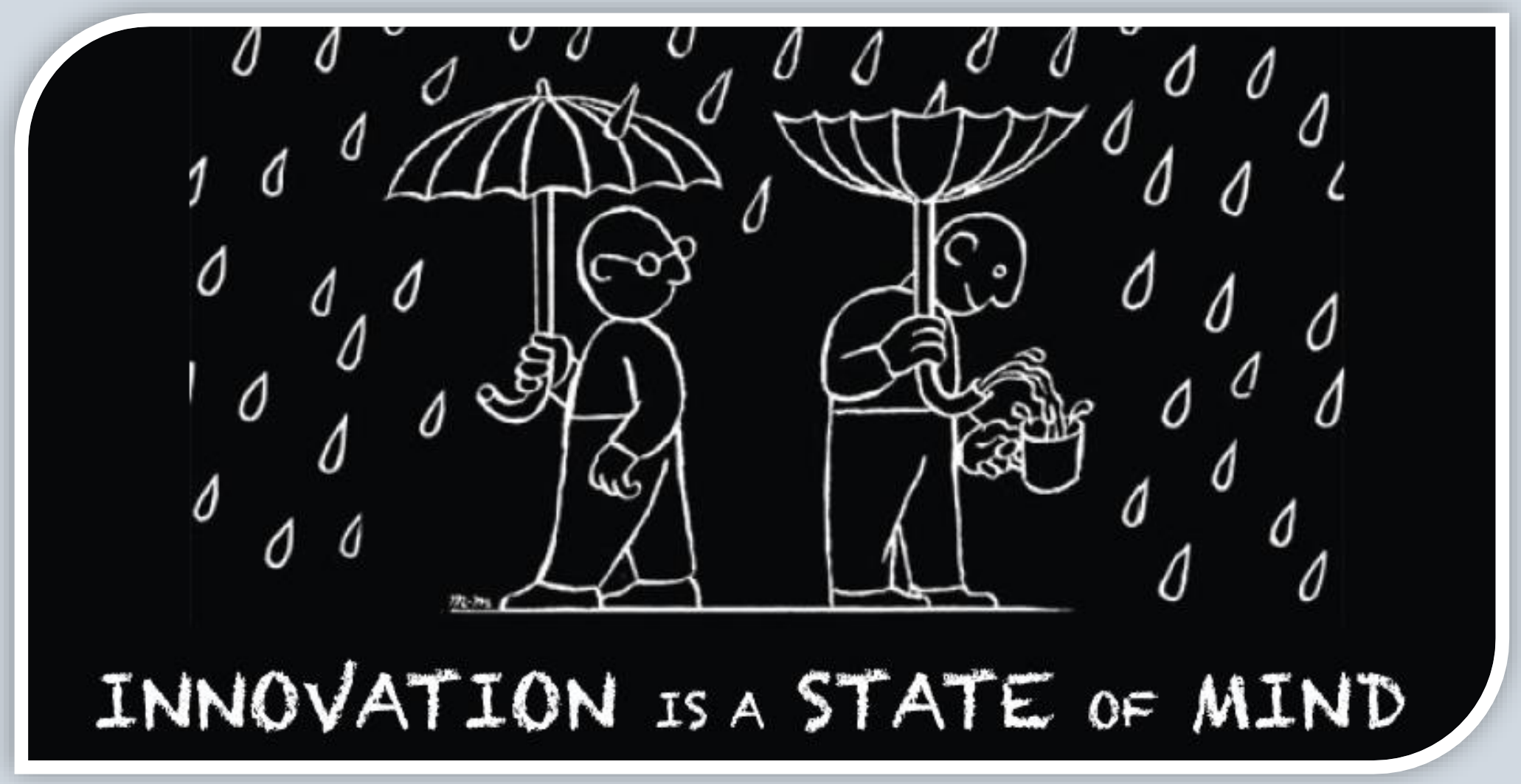
What is a Hackathon?

- aka Hack Day
- aka Code Night
- aka Codeathon

Hackathons as they grow ..



Why would you have one?



What they look like …

- ❑ Ideas and Sanity Checks
- ❑ Pitch Session
- ❑ Hacking!
- ❑ Final Presentations
- ❑ What Next?

Get Your Hack on !

Part 2

RWWA Hackathon Feb 2017

Testing in Hackathons

Unleash your Ideas

Doing More with Less

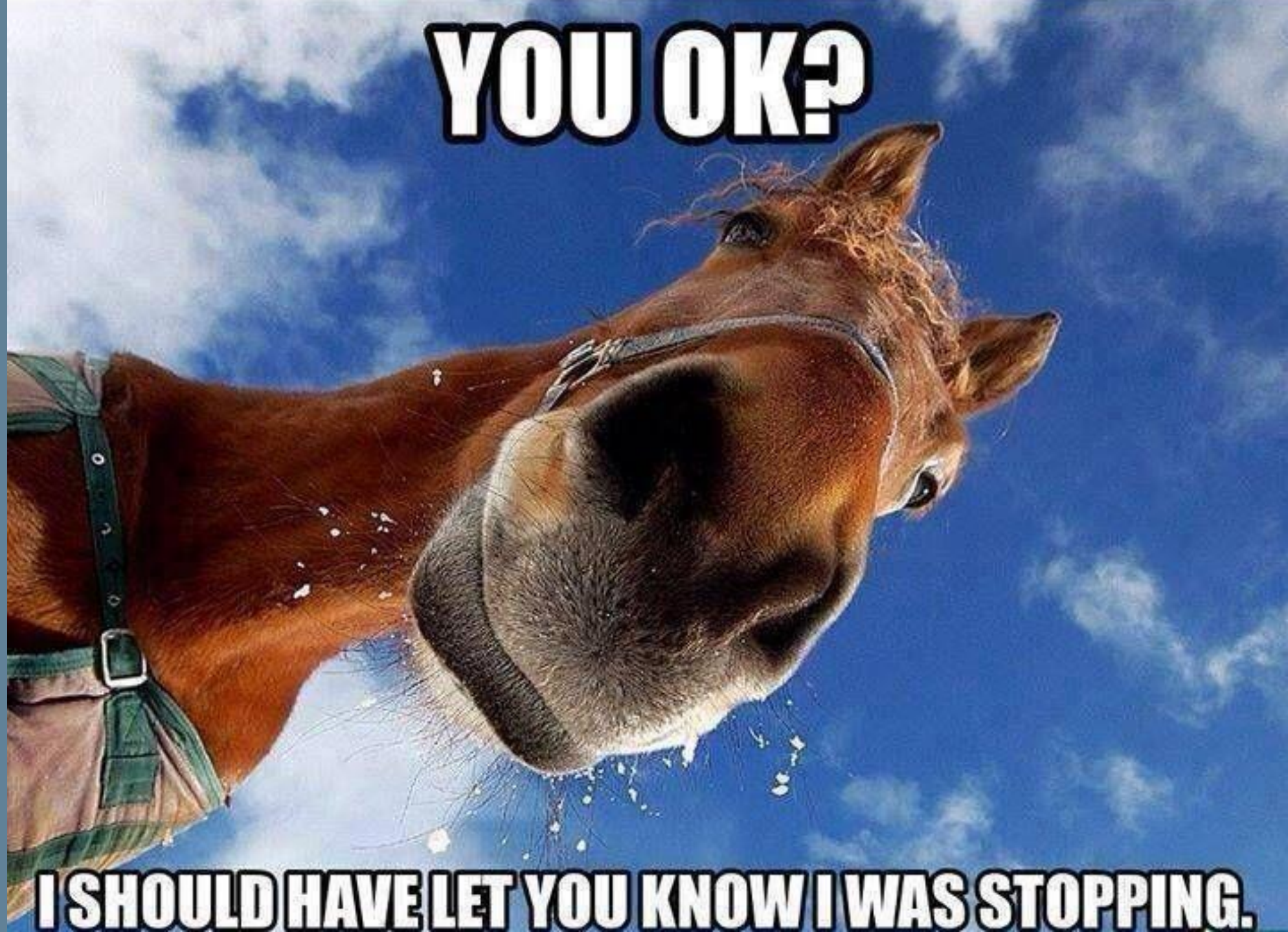
Concept Testing

Up Skill

Hackathons in Perth…

…and many others

Questions ...



Networking and Refreshments

We will resume at 18:45

Copyright © ANZTB

Presentation 2

Security-testing web applications in the Cloud

By Graham Weston

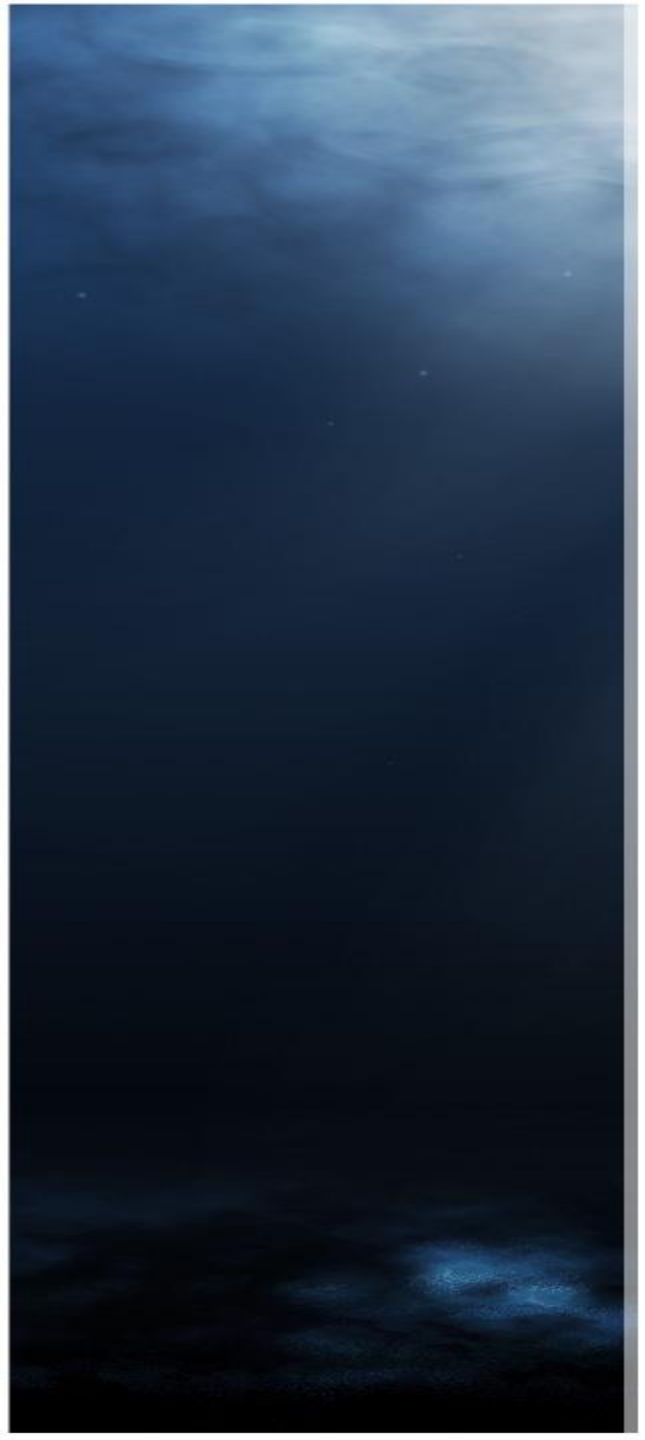
Copyright © ANZTB

Security Testing a Cloud-Based Web Application

Graham Weston

July 2017

© TemplatesWise.com



Agenda

Introduction and Background

Security Testing a Web Application

- Aims and objectives
- Workflow
- Static analysis and reviews
- Threat model
- Penetration test

Working in the Cloud

- Implications of testing on Cloud platforms
- Cloud risks
- Testing constraints

Tools, techniques and examples

ⓘ Health Warning

The tools and techniques discussed here are powerful:

- They should only be used under controlled conditions (and with the express consent of the owner of the system under test)
- Professional penetration testers are bound by strict codes of practise and ethics
- The distinction between security testing and hacking is entirely contextual

In short, don’t try this at home!

What is a Security Test?

Security testing verifies that a system is not vulnerable to malicious attacks.

It may comprise:

- Static analysis (code reviews, configuration reviews)
- Automated vulnerability scans
- Penetration test

Security testing is a type of non-functional testing

Key Security Principles

Security testing targets a number of key principles:

- Confidentiality
- Integrity
- Authentication
- Availability
- Non-repudiation

Threat Classification

OWASP STRIDE Threat Classification Model:

Category	Definition	Security Control
Spoofing	Threat action aimed to illegally access and use another user's credentials, such as username and password.	Authentication
Tampering	Threat action aimed to maliciously change/modify persistent data, such as persistent data in a database, and the alteration of data in transit between two computers over an open network, such as the Internet.	Integrity
Repudiation	Threat action aimed to perform illegal operations in a system that lacks the ability to trace the prohibited operations.	Non-Repudiation
Information disclosure	Threat action to read a file that one was not granted access to, or to read data in transit.	Confidentiality
Denial of Service	Threat aimed to deny access to valid users, such as by making a web server temporarily unavailable or unusable.	Availability
Elevation of privilege	Threat aimed to gain privileged access to resources for gaining unauthorized access to information or to compromise a system.	Authorization

Why do Security Testing?

To better understand and mitigate the risks posed to a system by external attackers:

- Data breach/theft (IPR, personal details etc.)
- Nefarious use of infrastructure/resources (via botnet etc.)
- Denial of Service

*"A **hacker** is someone who **seeks and exploits weaknesses in a computer system or computer network**. Hackers may be motivated by a multitude of reasons, such as profit, protest, or challenge."*

Why do Security Testing?

Data Theft:
Personal data
Business data
IPR

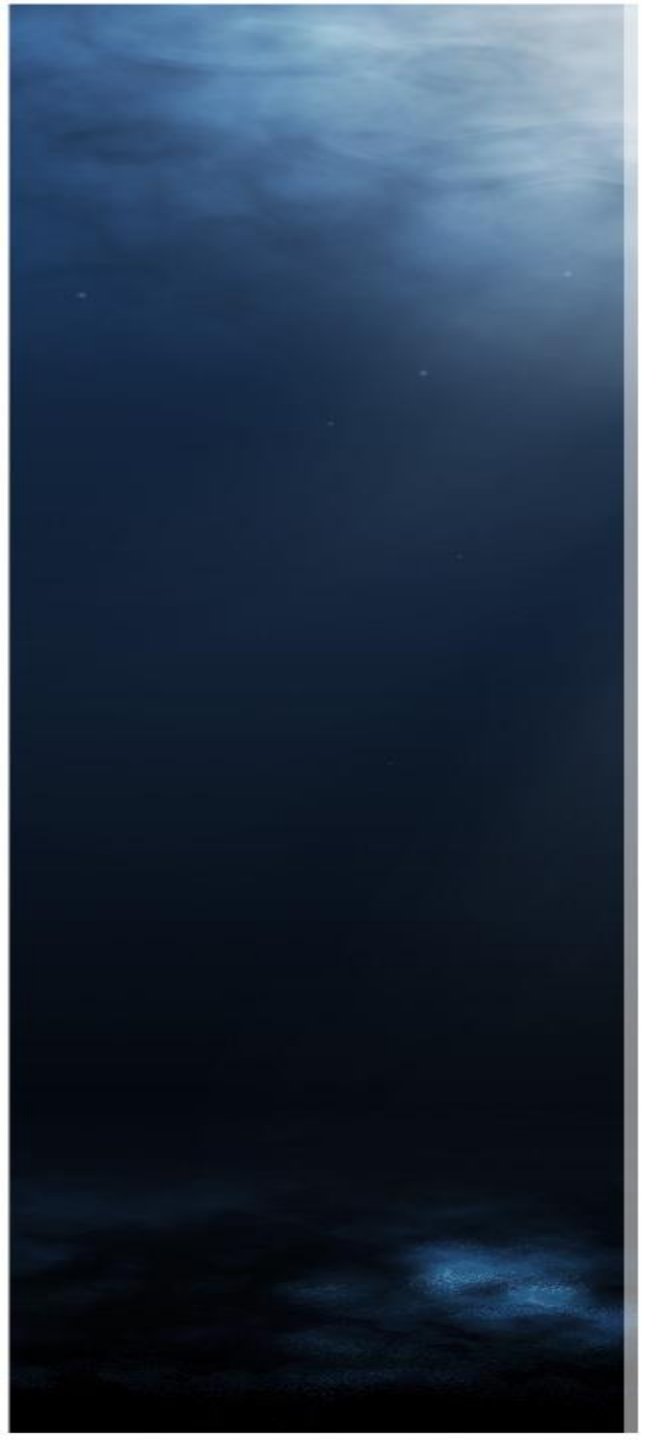
Reputation hijack:
Social media (Facebook, LinkedIn)
eBay/PayPal

Botnet:
DDoS traffic generator
Spam transmission
Click fraud

Hostage attacks:
Ransomware
e-mail account ransom
Webcam extortion

Credential Theft:
Online Banking
Social Media
Website credentials
Corporate e-mail

Virtual goods:
License keys
Online gaming assets, currency



Security Testing a Cloud-Based Web Application

Test Objectives

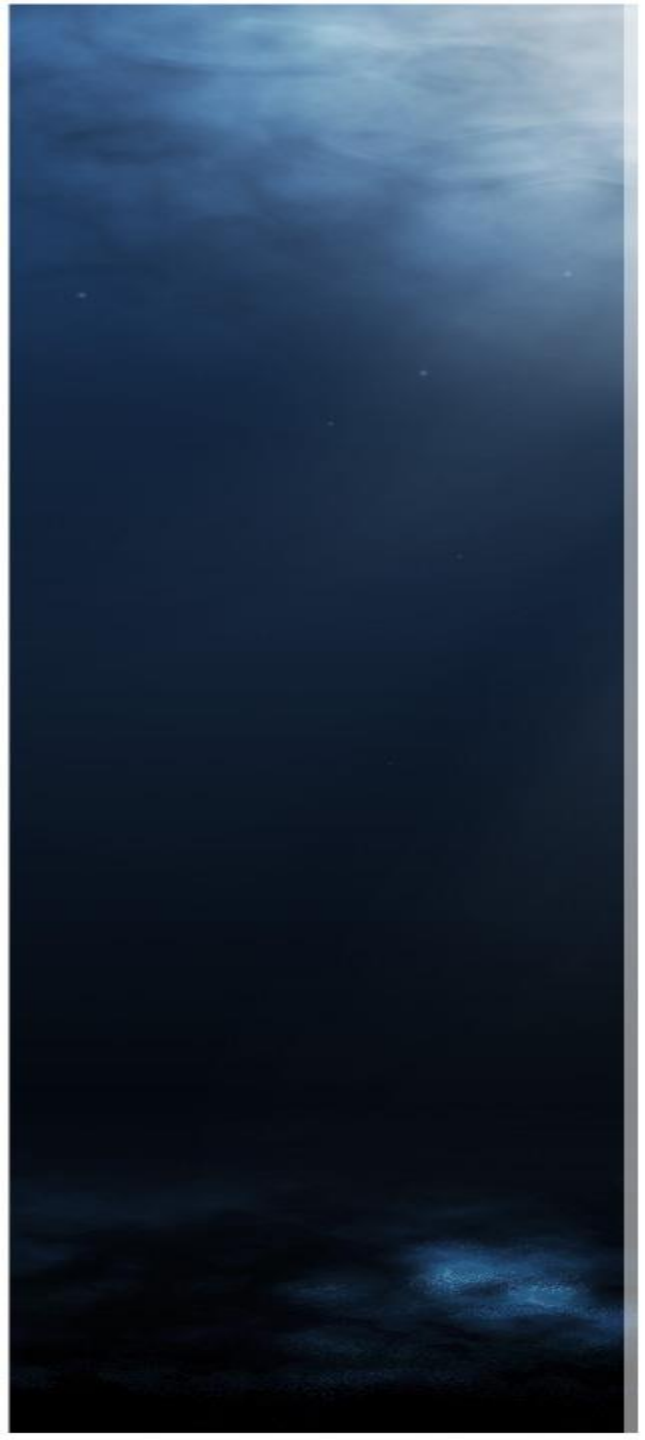
Example objectives for Security Testing Activity:

- Document a Security Test Plan
- Develop a Threat Model that documents the risk profile of the application
- Highlight defects/code issues early through code review
- Validate system configuration for application and CMS
- Carry out an external Penetration Test
- Validate DDoS mitigation measures

Working in the Cloud

We don’t control the infrastructure:

- Hosted on shared resources
  - Test or pre-PROD environments are hosted on the same infrastructure as PROD environments (and other people’s).
  - Some tests, particularly anything involving Denial of Service (DoS) could impact other tenants’ PROD systems.
- Providers’ goals are to maintain stability quality of service for their customers
- Cloud service Terms and Conditions generally impose constraint on security testing. For example, on Azure:
  - Need to notify providers of test schedule and activities
  - DoS/DDoS tests are not permitted



Test Execution - Tools and Techniques

Automated Tools

The good news:

- Commercial and open-source tools take away much of the hands-on complexity, using GUIs to drive the process
- Possible to deliver quick wins without coding, scripting, or a detailed understanding of the underlying technologies

The bad news:

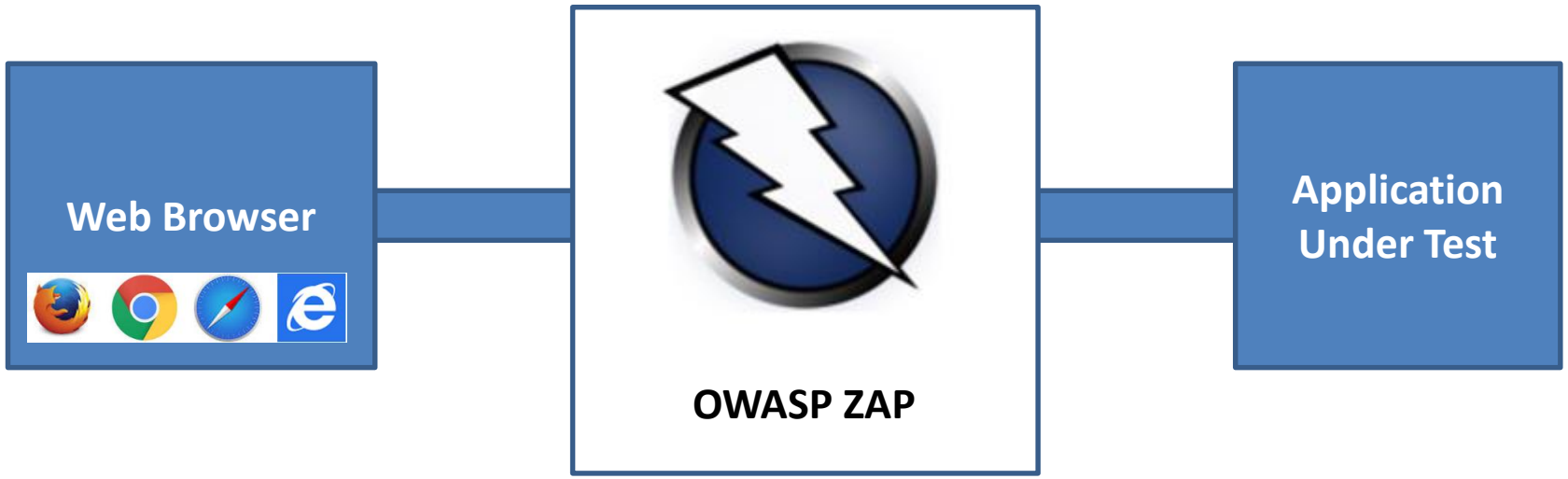
- They will only find well-known, existing vulnerabilities
- Finding complex/valuable vulnerabilities is a more labour-intensive process, dependent on a skilled pen-tester

OWASP ZAP

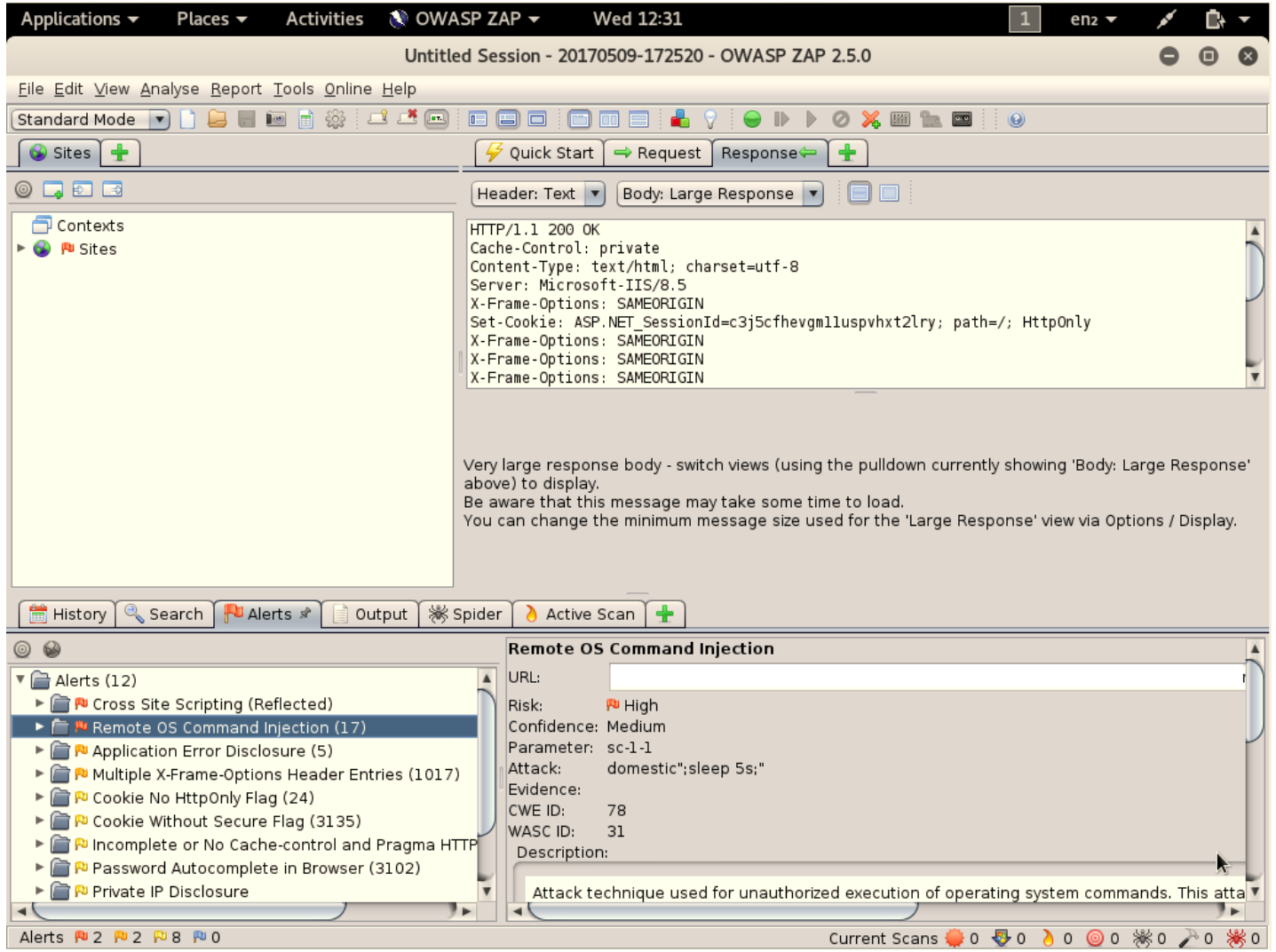
The OWASP Zed Attack Proxy (ZAP) is a popular free security tool:

- Actively maintained by hundreds of international volunteers.
- Can help automatically find security vulnerabilities in your web applications.
- A great tool for experienced pentesters to use for manual security testing.

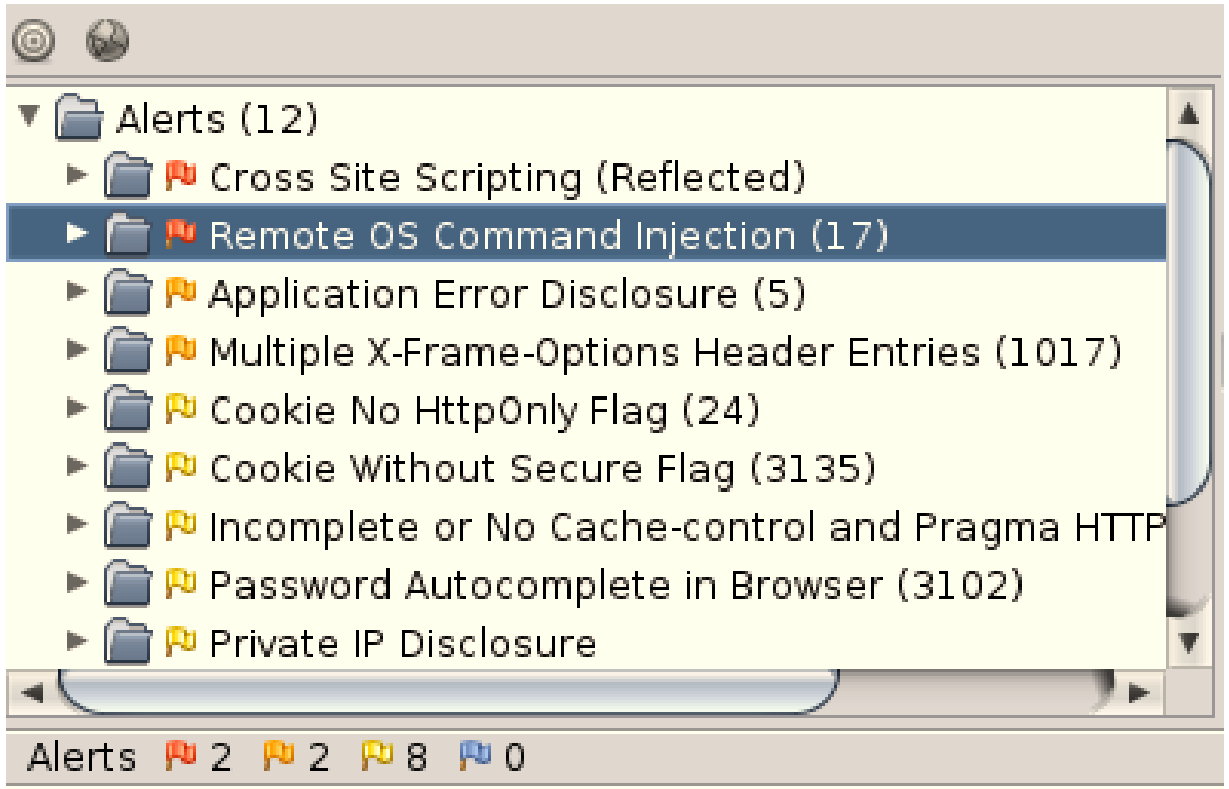
Deploying OWASP ZAP



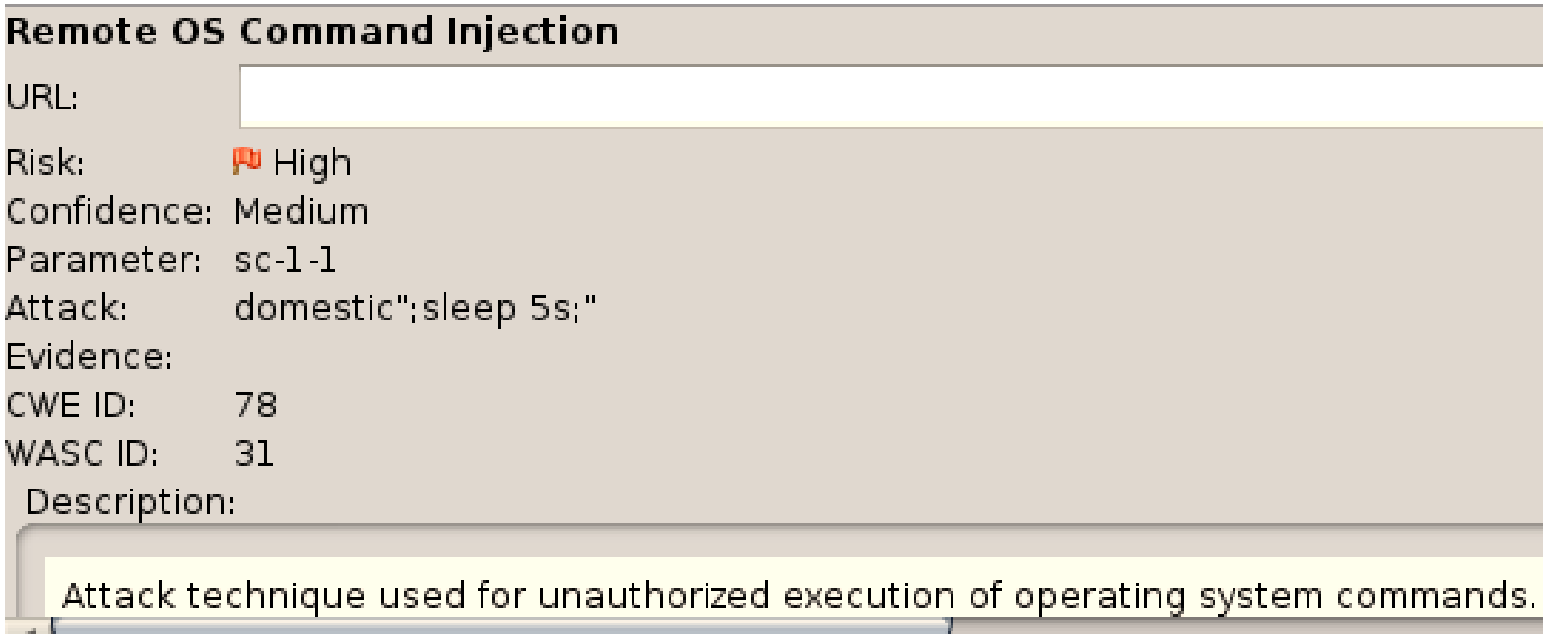
Sample ZAP Results



Sample ZAP Results (2)



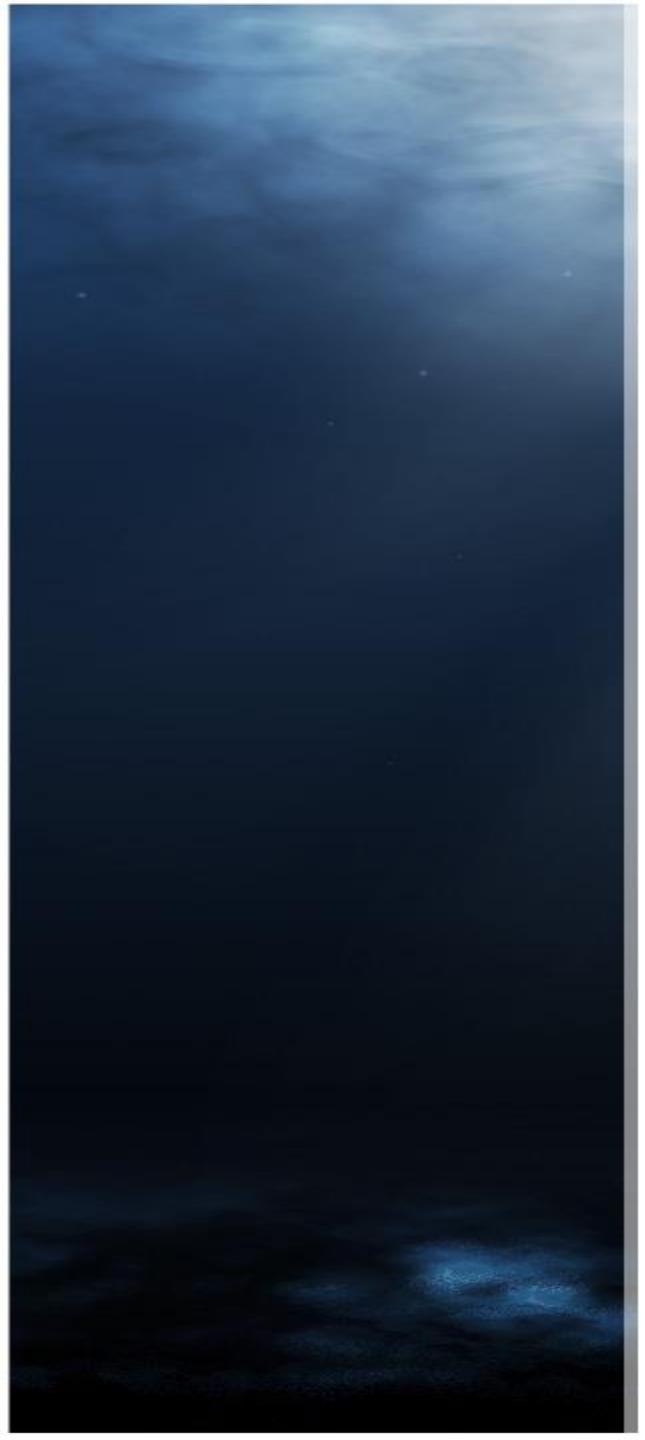
Sample ZAP Results (3)



Other Useful Tools

Kali Linux is a version of the Linux OS specially configured for carrying out security tests:

- Proxy/scanner tools
  - w3af (Web Application Attack and Audit Framework)
  - Burp Suite
- Wfuzz (fuzzing/brute forcing tool)
- Nmap (Network mapper and auditing tool)
- Browser developer tools/plugins
  - Library Detector (identifies JS libraries in use)



Common Vulnerability Types and Attacks

2017 OWASP Top 10

A list of the 10 Most Critical Web Application Security Risks:

- A1-Injection
- A2-Broken Authentication and Session Management
- A3-Cross-Site Scripting (XSS)
- A4-Broken Access Control
- A5-Security Misconfiguration
- A6-Sensitive Data Exposure
- A7-Insufficient Attack Protection
- A8-Cross-Site Request Forgery (CSRF)
- A9-Using Components with Known Vulnerabilities
- A10-Underprotected APIs

DDoS Attack

What is a DDoS attack?

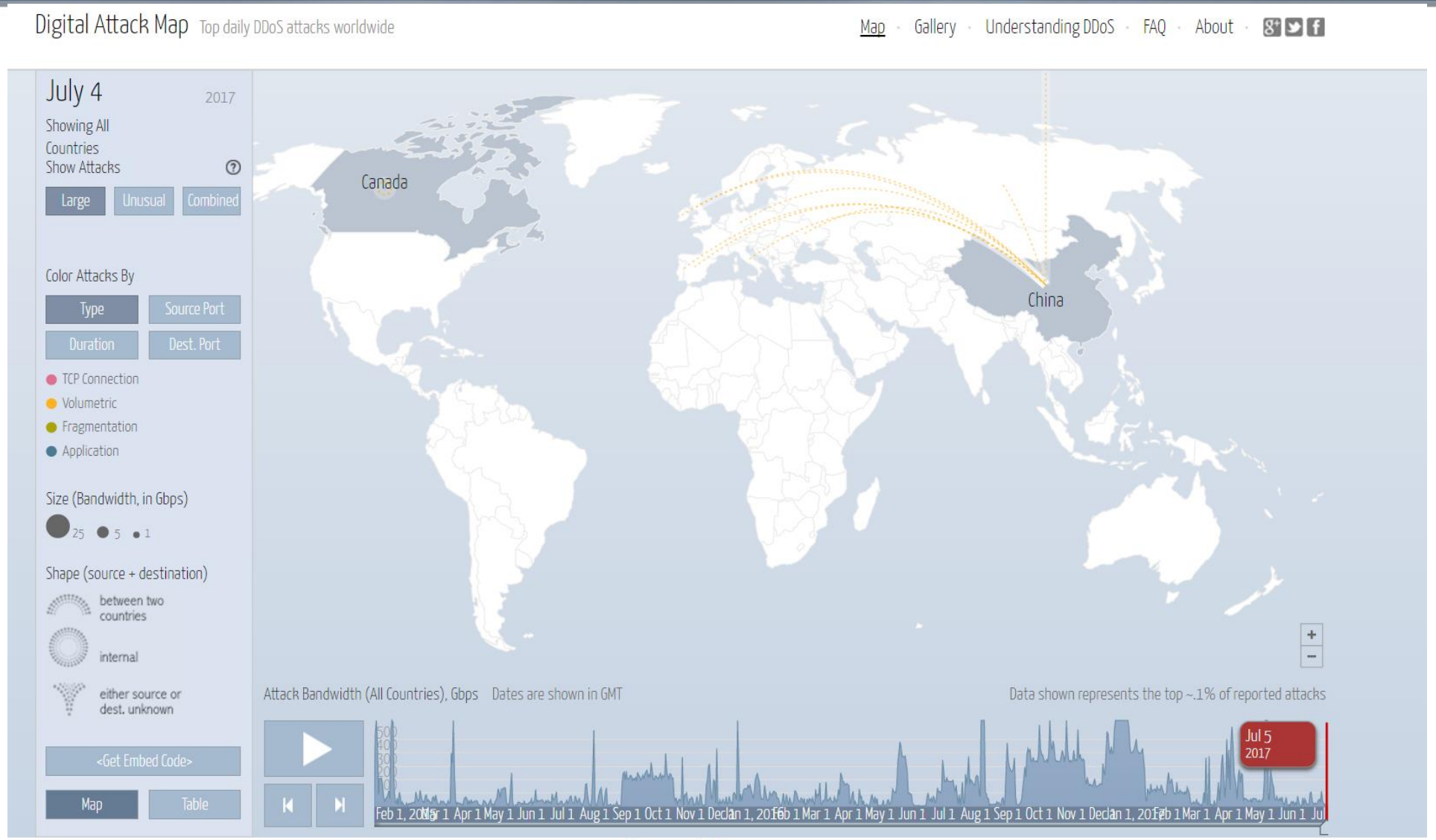
A network-borne Distributed Denial of Service attack renders a system unusable by consuming all of its bandwidth or resources with malicious traffic from multiple sources

Recent Examples:

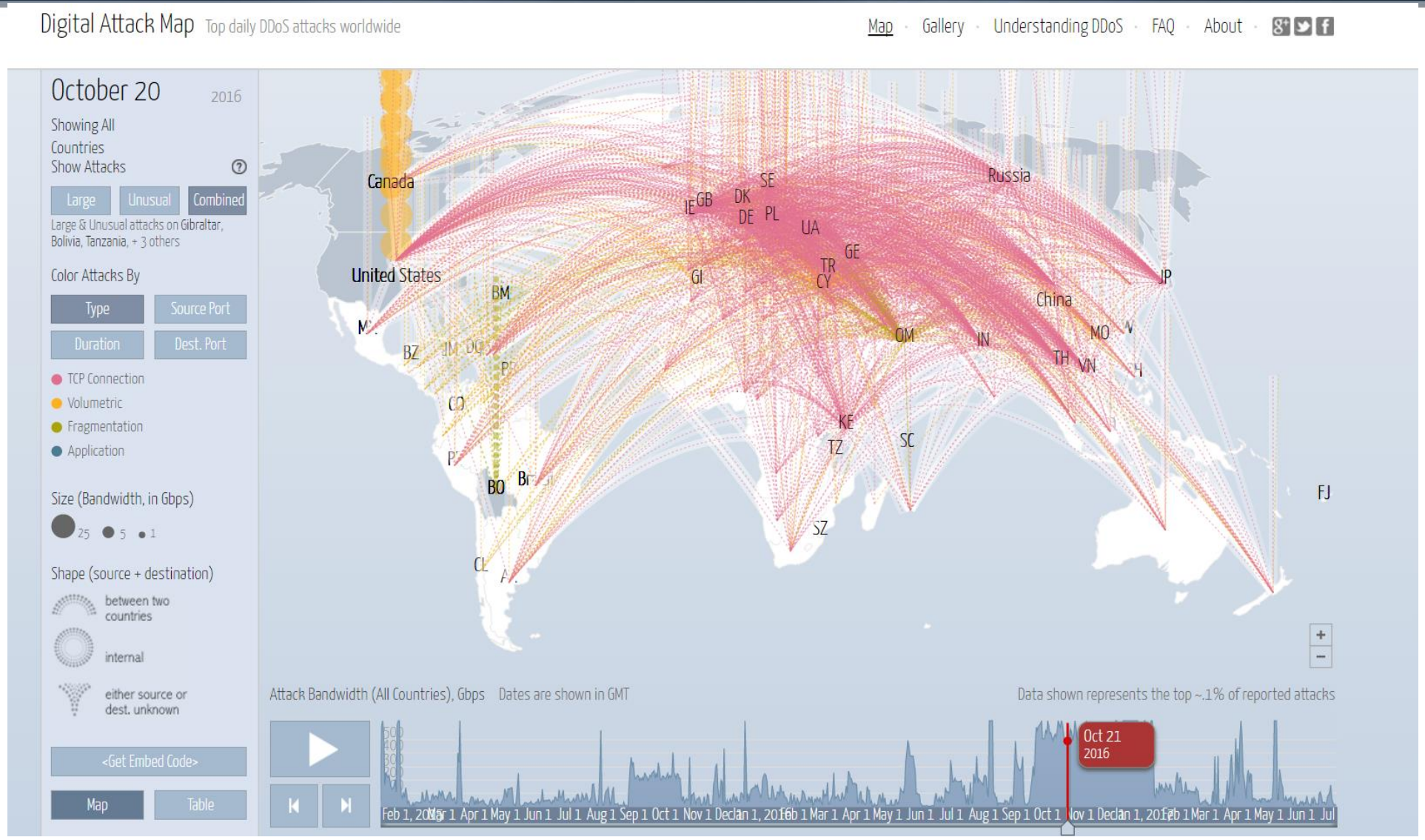
Significant attacks at end of 2016:

- 21/10/2016: Twitter, GitHub, Amazon, Netflix, Pinterest, Etsy, Reddit, PayPal, and AirBnb all taken off line for millions of users in a malicious attack by Mirai IoT botnet
- 14/11/2016: Major attack on 5 Russian Banks

DDoS Attack Traffic Visualisation



DDoS Attack Traffic Visualisation



Types of DDoS Attack

How does it work?

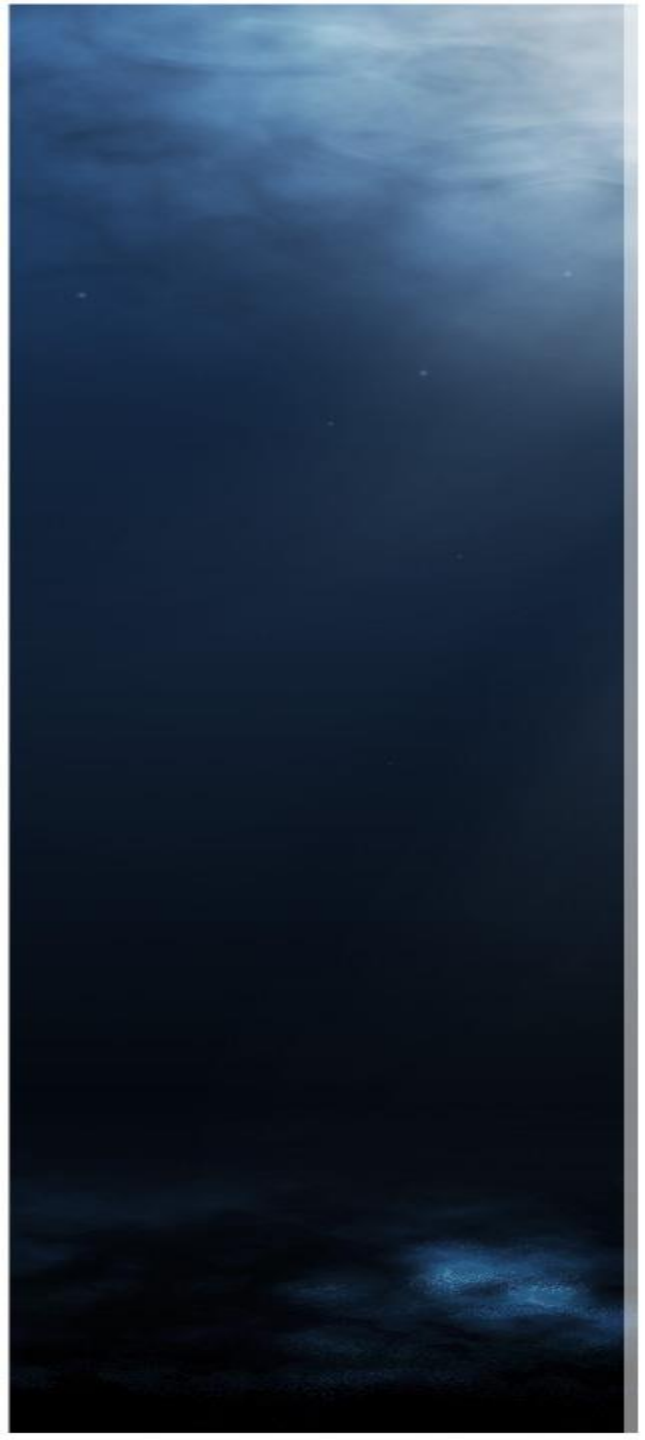
- Traffic flood (UDP, HTTP, ICMP)
- Protocol-based
  - Slow Loris (initiates a TCP connection then stops and leaves session hanging)
  - SYN flood
- Malformed packets
  - Packet fragmentation
  - Ping of Death (oversized ICMP packets)

DDoS Attack

How to mitigate DDoS attacks:

Non-trivial, especially for sophisticated protocol-based attacks.

- Monitoring: look for patterns in traffic
  - Understand what normal traffic looks like
  - Identify changes that indicate potential DDoS
- Cloud providers offer anti-DDoS technology (e.g. CloudFlare, Akamai) and anti-DDoS emergency response services.



Useful Resources and Further Reading

Useful Resources

OWASP (Open Web Application Security Project)

www.owasp.org

OSSTMM (Open Source Security Testing Methodology Manual)

www.osstmm.org

NIST (National Institute of Standards and Technology)

csrc.nist.gov/publications/nistpubs/800-115/SP800-115.pdf

DDoS Attack Map

http://www.digitalattackmap.com/

Conclusions

Conclusions

- Effective security testing comprises a series of activities (static analysis including configuration and code reviews, not just a pen-test).
- Testing systems hosted in the Cloud constrains test execution
  - Need to notify provider of timing and nature of test activities
  - Some types of test, e.g. DDoS, are prohibited entirely
- There are standards, such as OWASP, which provide useful but not exhaustive resources.
- Open-source tools and commercial tools are available that offer a range of useful functionality:
  - Automating common and labour intensive tasks
  - Identifying the attack surface of the application under test
  - Finding, tracking and managing vulnerabilities over the product lifecycle
  - Integration with CI tools to automate security scans as part of build/deployment process