Digital Interruption

Automating myself out of a job

A pen tester’s guide to left shifting security testing

+44 (0)161-820-3056

www.digitalinterruption.com

jahmel@digitalinterruption.com

DIGITAL INTERRUPTION


DEVELOPERS
DESIGNERS
PROJECT MANAGERS
QA
SYSADMINS
SECURITY
SEEN BY DEVELOPERS
SEEN BY DESIGNERS
SEEN BY PROJECT MANAGERS
SEEN BY QA
SEEN BY SYSADMINS
SEEN BY SECURITY

"We need a pen test"

"Pen testing Sucks" – Pen Testers

- Super boring to test most applications
- Have to report low risk issues
- No one fixes the issues anyway

"Pen testing sucks" - Developers

- Reports padded with low risk issues
- Pen testers don't understand the context of vulnerabilities
- Too much ego
- We can't stop development/releasing comes first

WHO AM I?

- Jahmel Harris
- Pen Tester/Security Researcher at Digital Interruption (@DI_SECURITY)
- Mobile | Radio | Reverse Engineering

- @JayHarris_Sec
- @mcrgreyhats
- @DI_Security

Vincent Yiu @vysecurity · Jun 4

Red tip #86: Red team and attack simulation is not penetration testing. You shouldn't be really testing anything, but simply infiltrating.

1 4 11

Jay Harris @JayHarris_Sec · Jun 4

Way to make terms more confusing 😜 if not testing, what use is red teaming? PT != vuln assessment but RT and PT are pretty interchangeable.

3 1

Vincent Yiu @vysecurity · Jun 4

RT and PT aren't interchangeable... PT is vuln assessment. If PT isn't vuln assessment then what is vuln assessment?

2

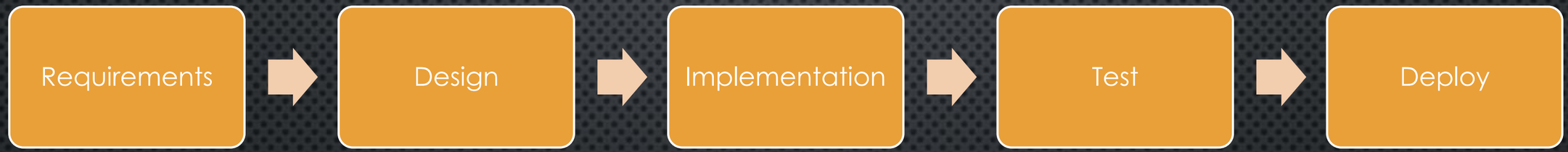
Jay Harris
@JayHarris_Sec

Replying to @vysecurity

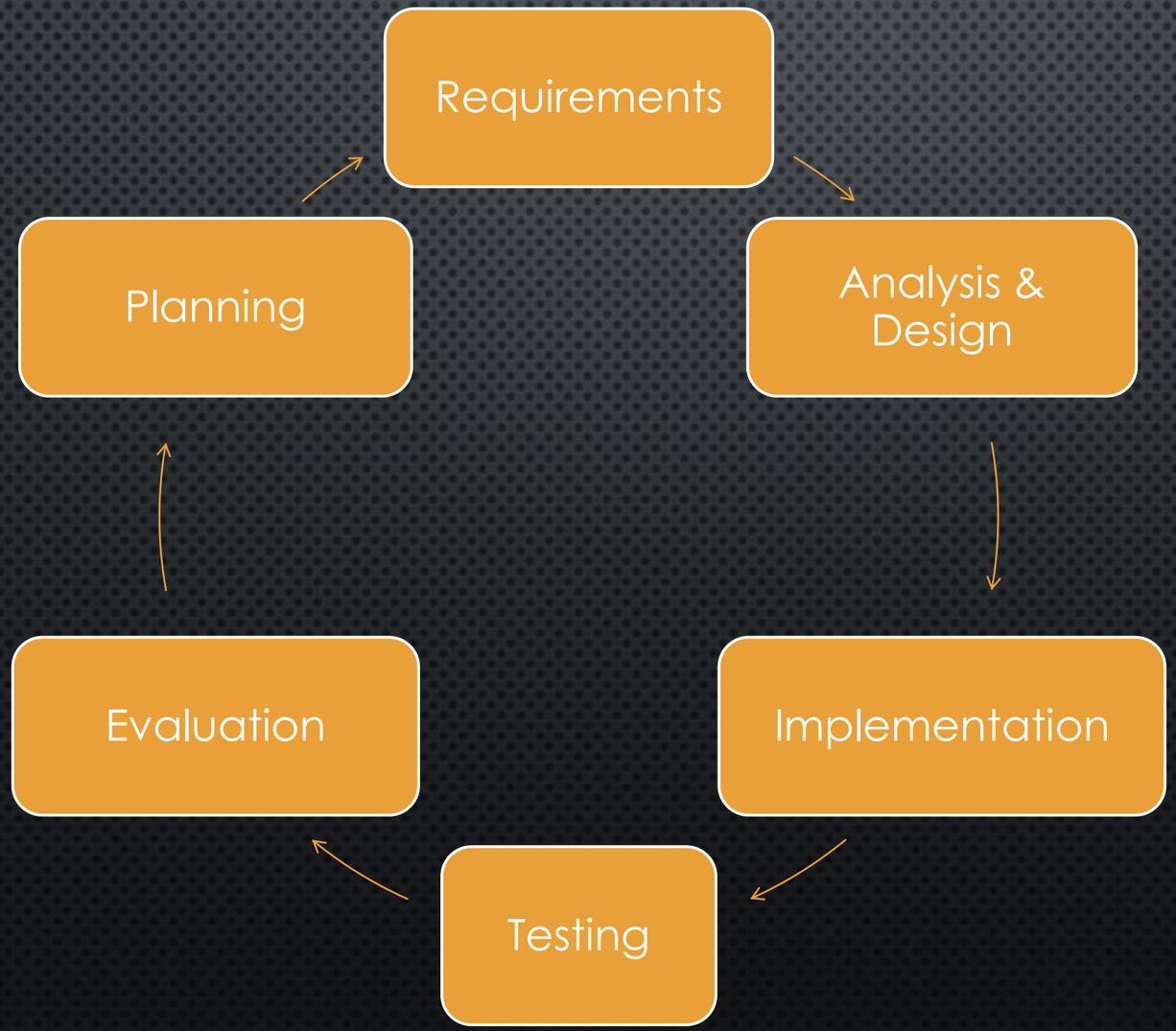
If the infosec community can't agree, no wonder our clients are confused! VA is VA. Pen testing is taking the role of an attacker.

12:32 PM - 4 Jun 2017

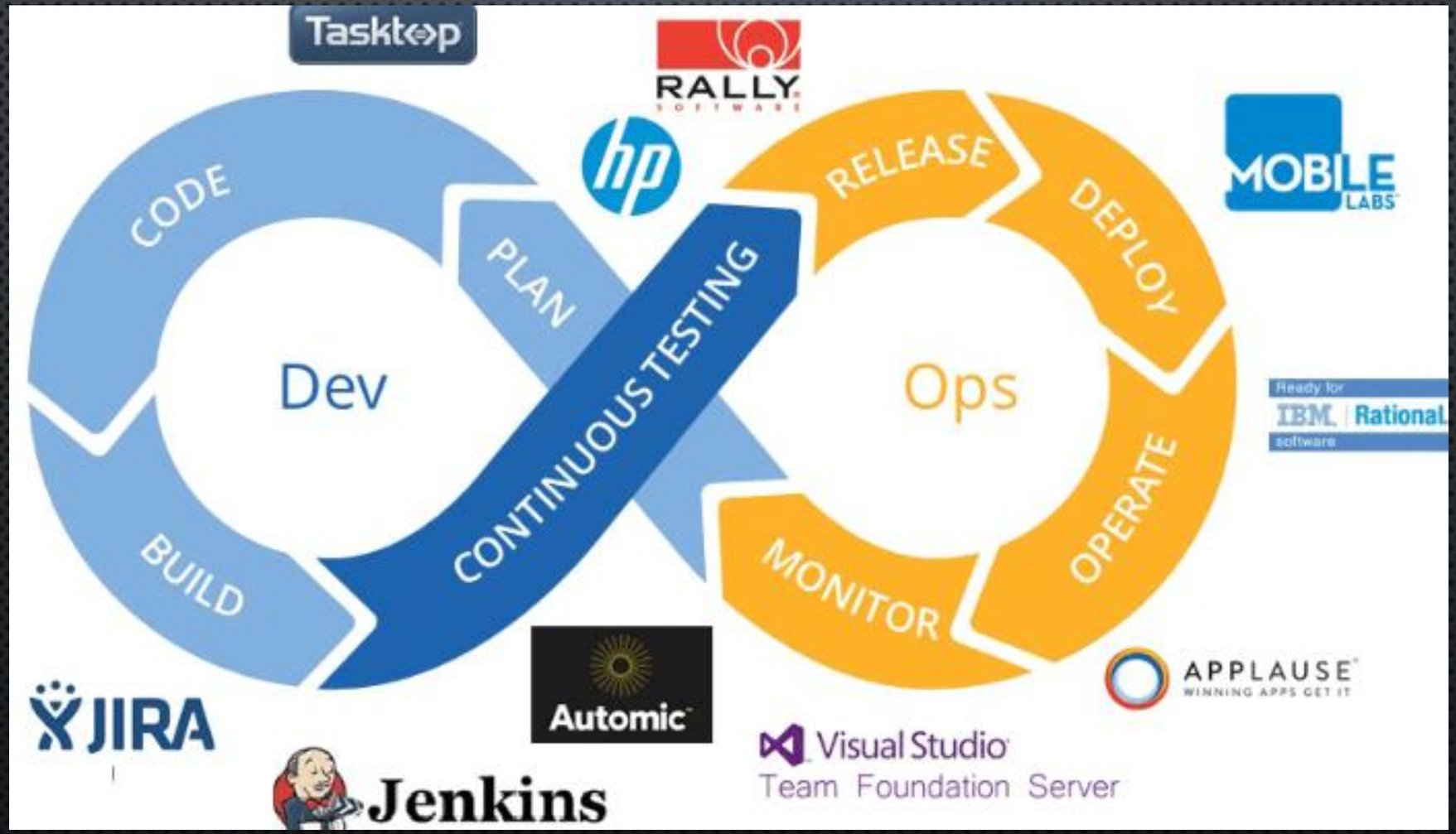
"TRADITIONAL" DEVELOPMENT



AGILE DEVELOPMENT



CI/CD - DEVOPS



https://media.licdn.com/mpr/mpr/AAEAAQAAAAAAAAU-AAAAJGM0OTM0M2Q4LTNjZDYtNDA0ZC05YjEwLWRlNzAxZDYxN2MyNQ.png

TRADITIONAL PEN TEST

Requirements → Design → Implementation → Test → Deploy

Traditional Pen Test

- Turn up
- Wait for requirements
- Test
- Wait for requirements
- Test
- Leave
- Write Report
- Send report

@JayHarris_Sec

@JayHarris_Sec

@JayHarris_Sec

APP DESCRIPTION (CASE STUDY)

- This app (Android/iOS) is used to make voip calls
- Made up of the clients and web service
- The pen test done from multiple perspectives and a subset shown here

VULNERABILITIES - HIGH

- Mobile Verification Code is Susceptible to Brute-Force Attacks
- Possible To View Other User's Messages
- SSL Validation Disabled
- Directory Traversal In Web Service

VULNERABILITIES - MEDIUM

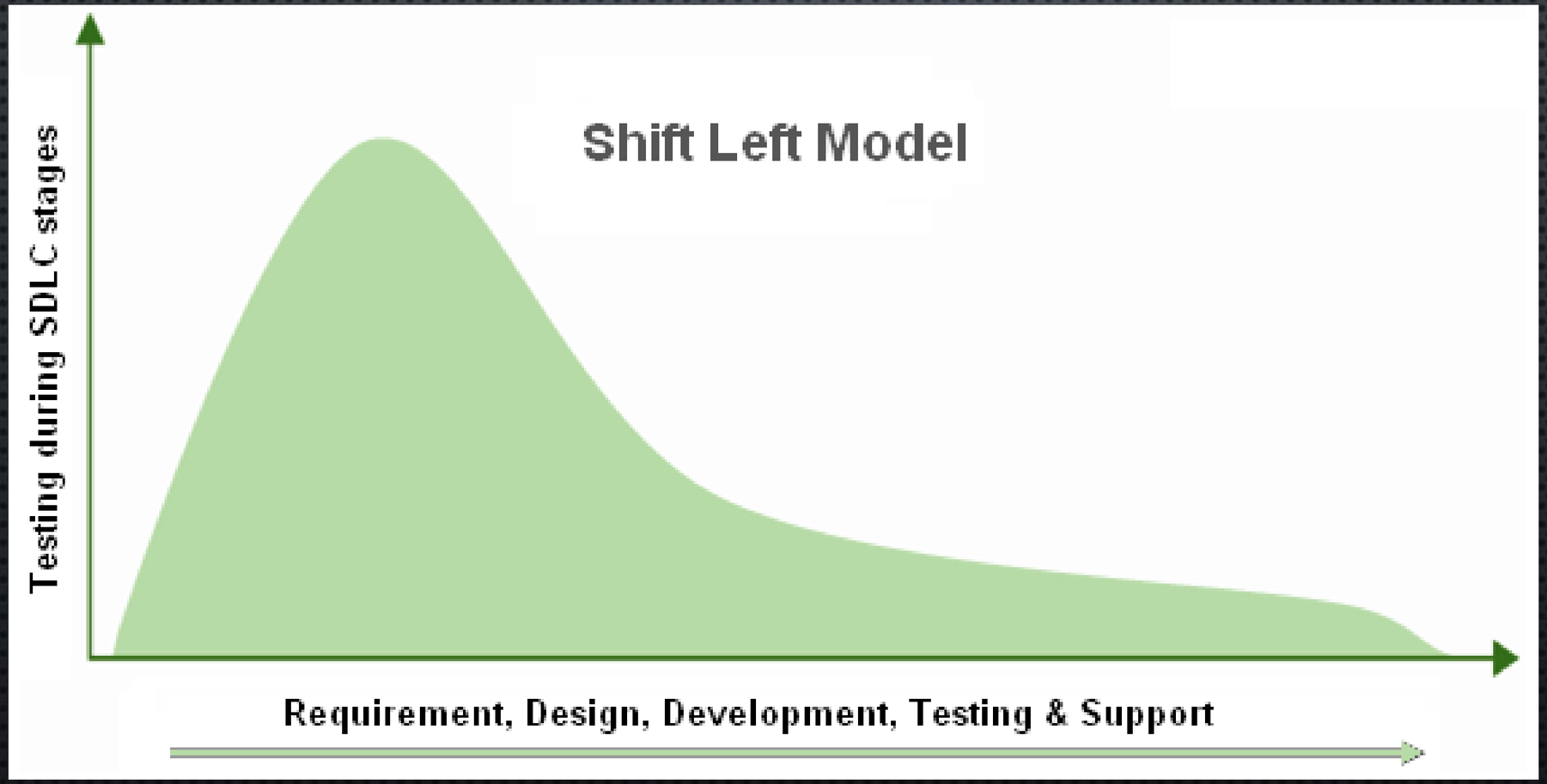
- Backups Allowed
- No Permissions on Android IPC
- Sensitive Information Stored In Application Sandbox
- SQL Injection in Android Content Provider
- Weak SSL/TLS Ciphers Supported
- Weak Authentication in Applications

VULNERABILITIES - LOW

- Apple Transport Security weakened
- Excessive Logging
- Lack of Anti-Debugging
- Version Banner Disclosure
- Lack of root detection

GO LIVE?

SHIFT LEFT SECURITY TESTING



http://blogs.bmc.com/wp-content/uploads/2017/07/Shift-Left-Model.png

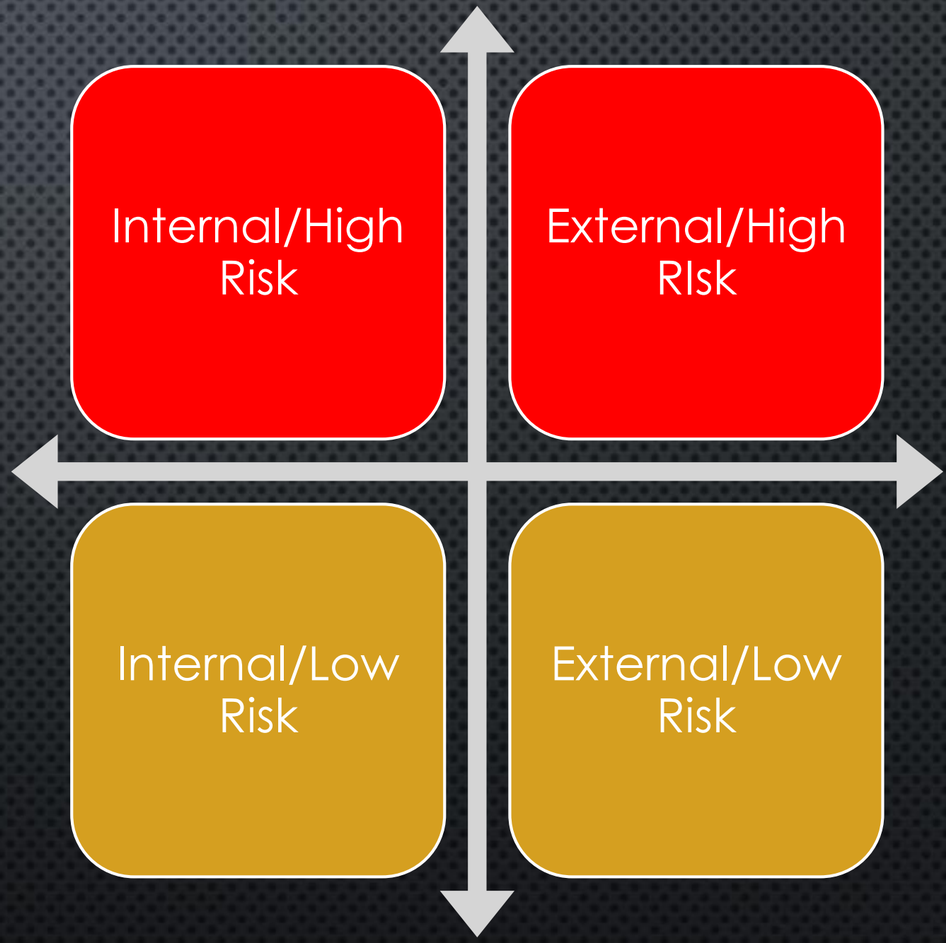
REQUIREMENTS GATHERING

RISK RATING

- THINKING LIKE AN ATTACKER – WHATEVER WORKS
  - THREAT MODELLING
  - PREVIOUS PEN TEST REPORTS
  - ATTACK TREES
  - EXTERNAL HELP

RISK RATING

- AN INTERNAL BROSUREWARE APPLICATION – PFFFFFFT WHO CARES? (UNLESS WE DO)
- FINTECH APPLICATION PROBABLY NEED TO CONSIDER SECURITY



RISK RATING

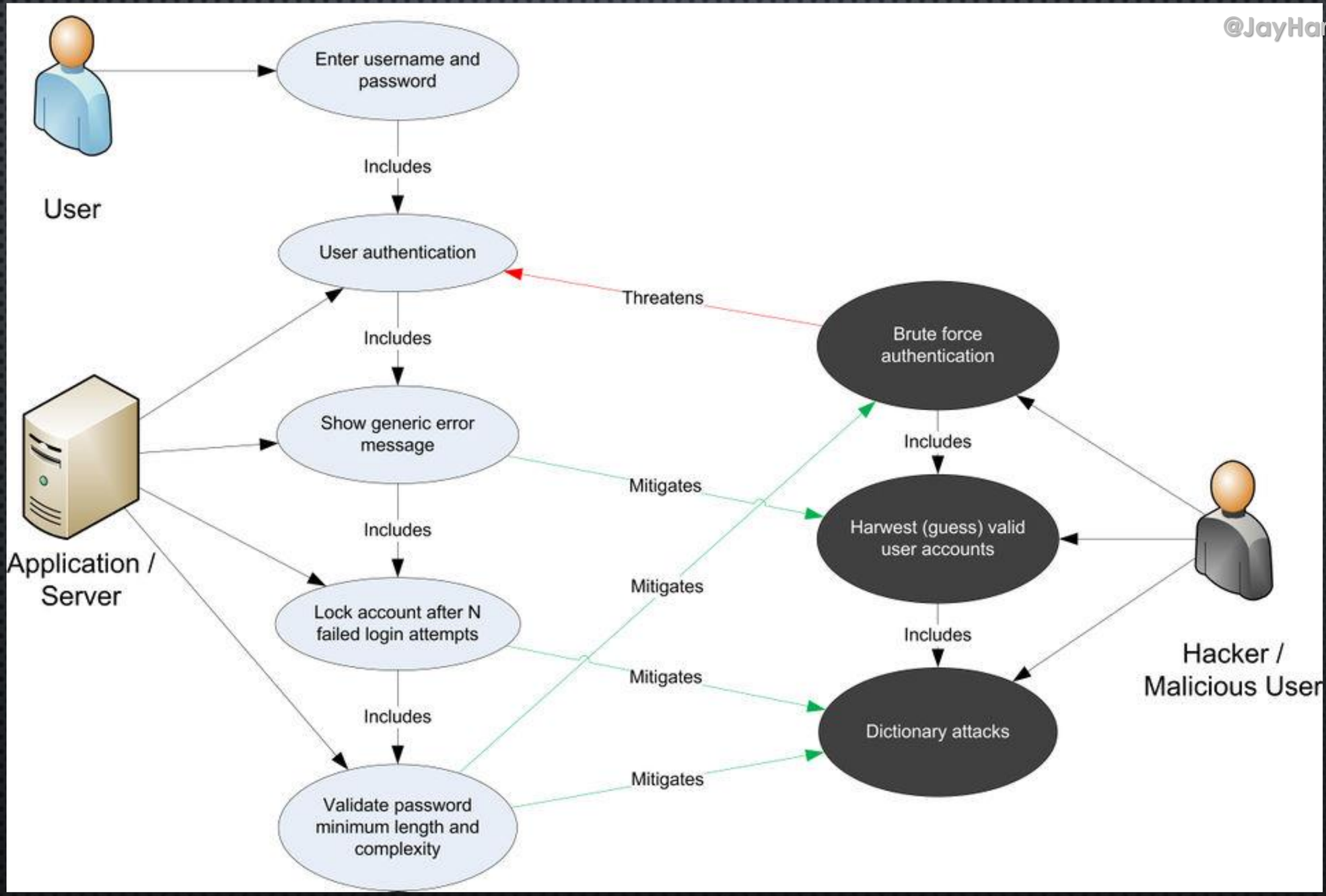
- A user requires knowledge of a username and password to use the application
- A user requires knowledge of their username and password to view their messages
- Sensitive data should not leave the application sandbox
- Sensitive data should be encrypted
- A user requires the MVC to successfully register
- Sensitive data should not be logged
- The application should not run on a rooted device*

Risk Rating – Abuse Stories

- As an attacker, I want to log into the application without knowing the password
- As an Attacker, I want to read files in the application sandbox
- As an attacker, I want to run the application on a rooted device for further vulnerability assessment
- As an attacker, I want to reverse engineer the application in order to perform further vulnerability assessment

Risk Rating – Abuse Stories

- As an attacker, I want to log into the mobile application without knowing the password
  - By brute forcing the password
  - Launching the activity manually
  - SQL Injection
  - ???



SECURITY REQUIREMENTS

- HELP US TO FIND AND ADDRESS SOME DESIGN ISSUES

DEVELOPMENT

Embedding security knowledge in the team

- Training
- Pairing
- Security SME
- Security Champion
- Security code review
- "Chat Ops"

UNIT TESTING

- DEVELOPERS KNOW HOW TO TEST THEIR CODE
- LET'S WRITE SOME UNIT TESTS (SOMEONE ELSE ON THE TEAM)
  - TDD/BDD

UNIT TESTING

Given an attacker can submit a username and password

When they try more than 5 incorrect passwords

Then the account should be locked

```
@Given("an attacker can submit a username and password")
public void AppIsRunning(int width, int height) {
    login = new Authenticator();
}

@When("they try more than 5 incorrect passwords")
public void incorrectPasswordIsTriedMoreThan5Times() {
    for(int i=0;i<=6;++i)
    {
        login.authenticateUser(user,"password")
    }
}

@Then("The account should be locked")
public void theAccountShouldBeLocked(String grid) {
    assertThat(login.accountLocked(username), true));
}
```

VULNERABILITIES - HIGH

- ~~Mobile Verification Code is Susceptible to Brute-Force Attacks~~
- Possible To View Other User's Messages
- SSL Validation Disabled
- ~~Directory Traversal In Web Service~~

VULNERABILITIES - MEDIUM

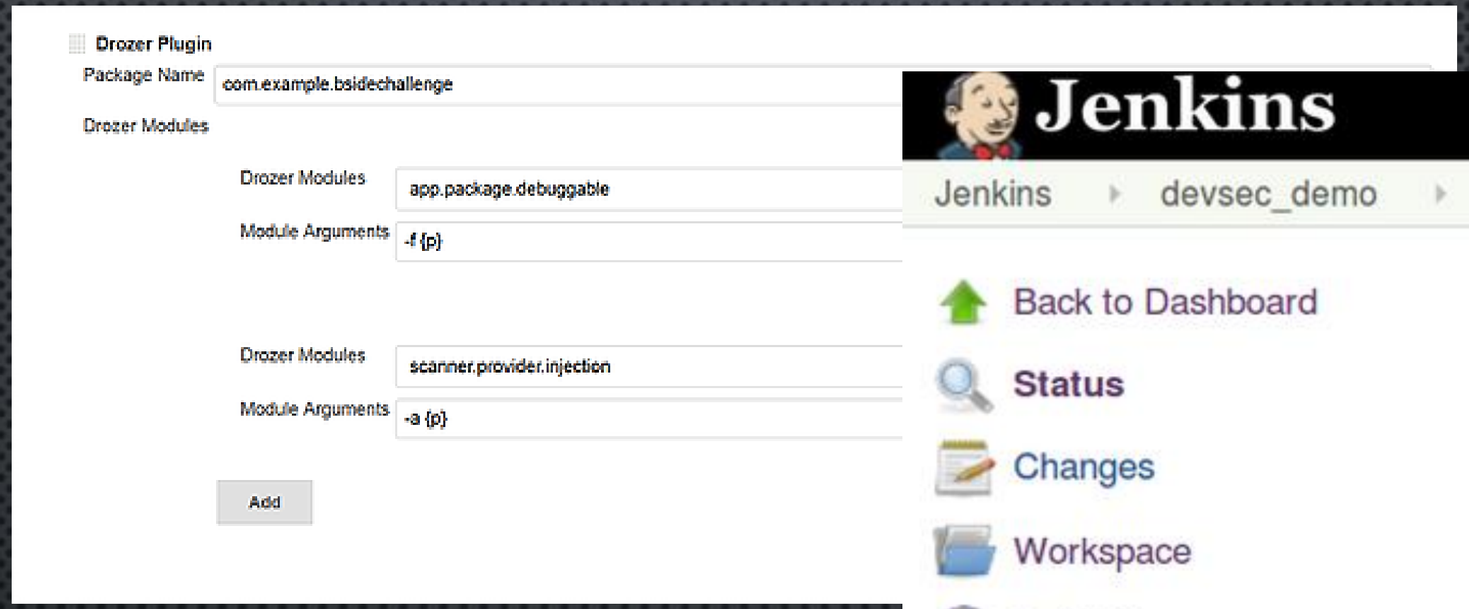
- Backups Allowed
- No Permissions on Android IPC
- Sensitive Information Stored In Application Sandbox
- ~~SQL Injection in Android Content Provider~~
- Weak SSL/TLS Ciphers Supported
- Weak Authentication in Applications

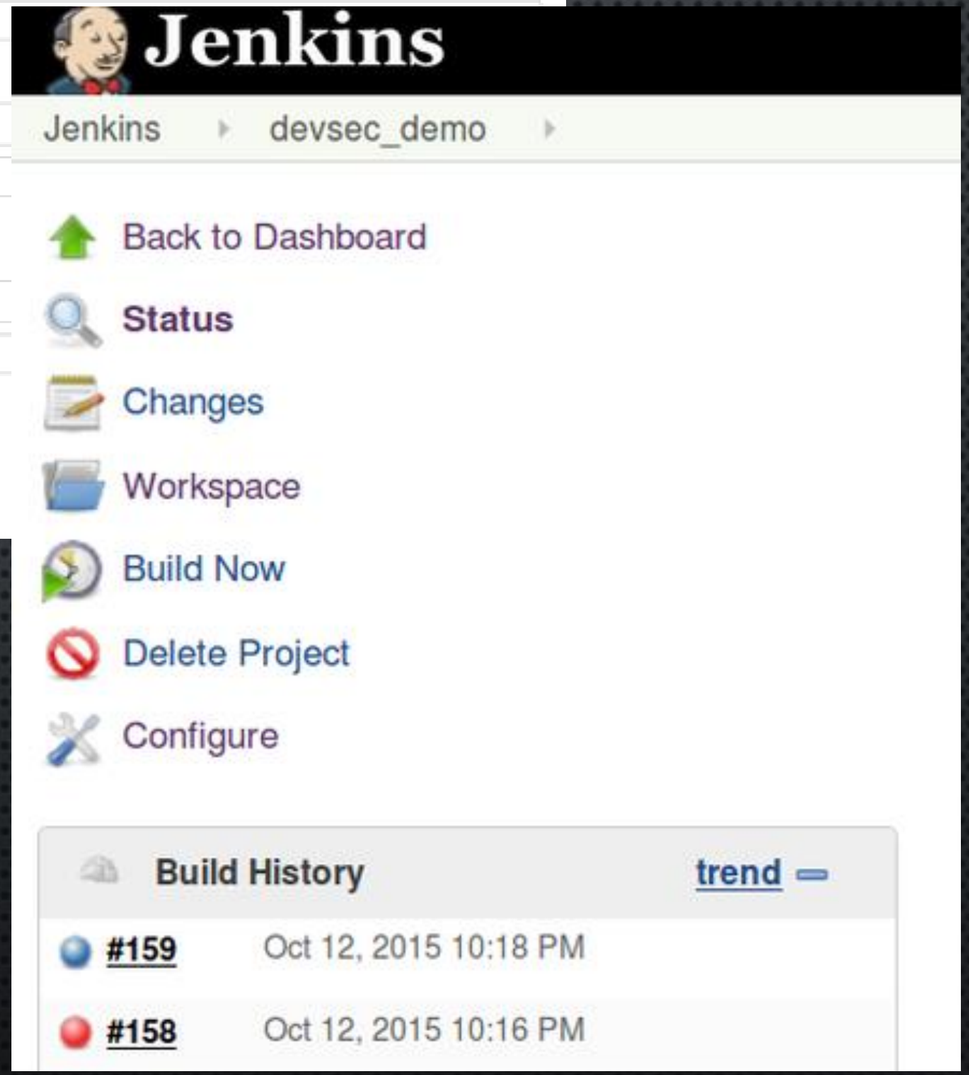
VULNERABILITIES - LOW

- Apple Transport Security weakened
- Excessive Logging
- Lack of Anti-Debugging
- Version Banner Disclosure
- Lack of root detection

SECURITY TOOLING

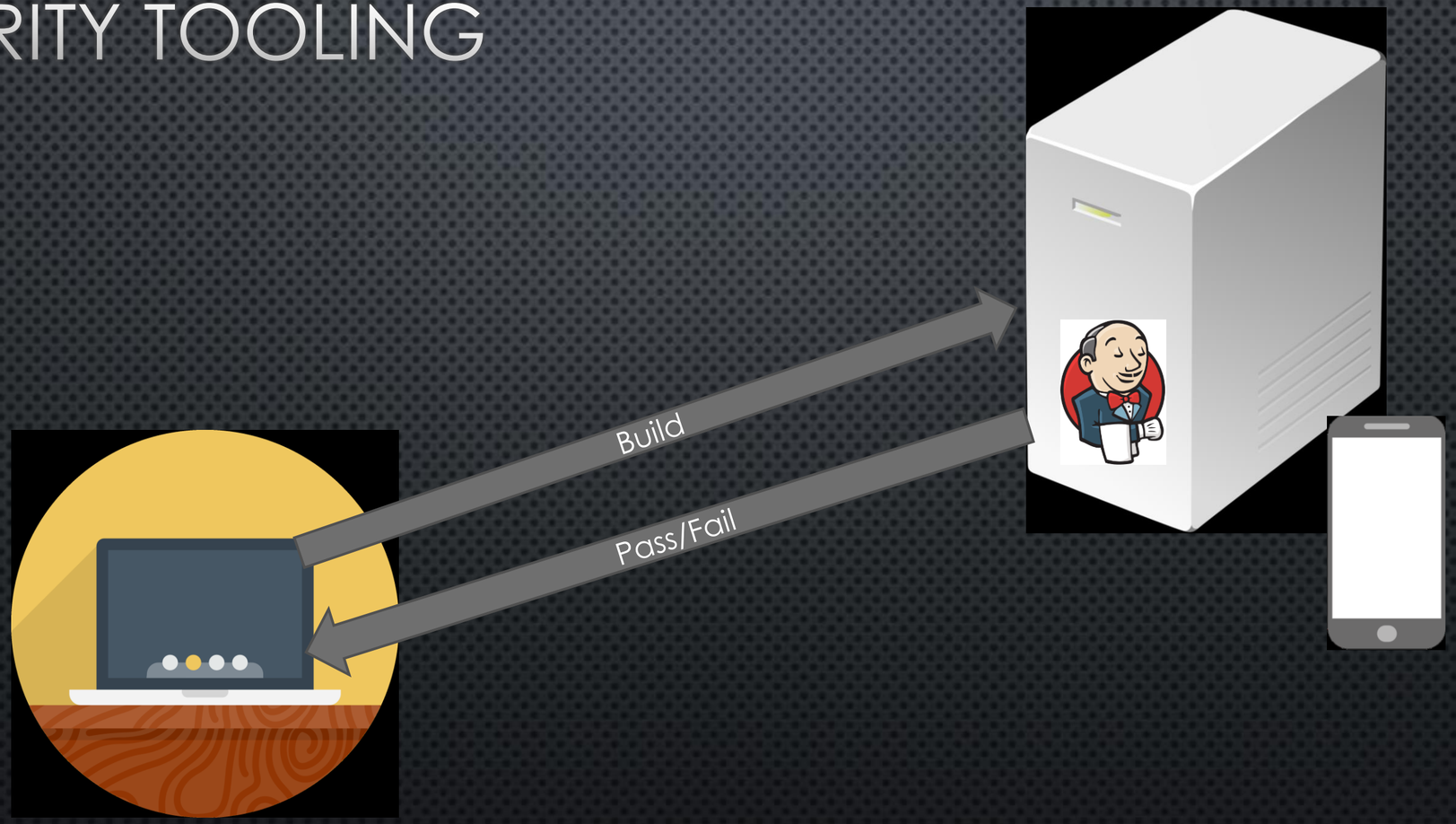
- NO MORE HACKER TOOLS FOR HACKERS (PLEASE!)
- TOOLS FOR DEVELOPERS AND TESTERS
  - MAKE OUR OWN (OR AT LEAST WRAPPERS)

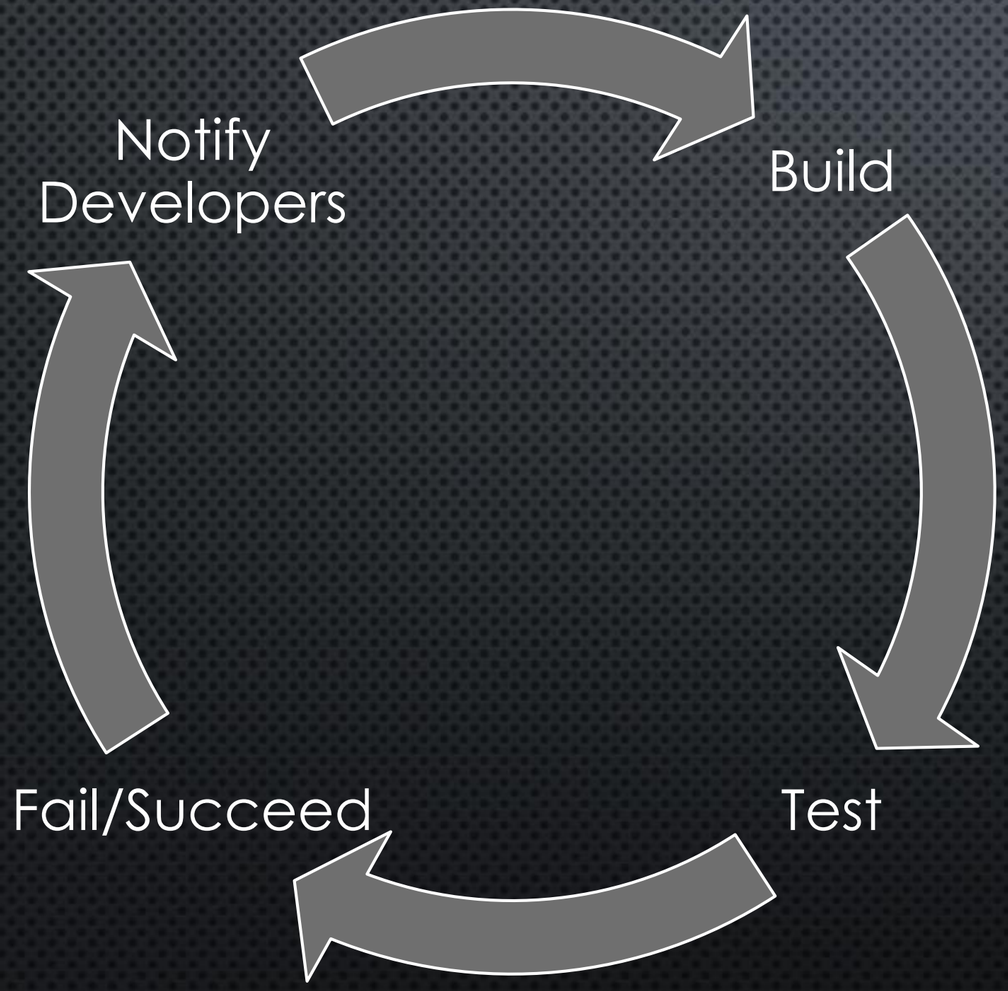




http://bit.ly/2eNQ9Pr

SECURITY TOOLING




Notify Developers
Build
Test
Fail/Succeed

SECURITY TOOLING - SAST

- SOURCE CODE SCANNING
- PATTERN MATCHING
- TAINT ANALYSIS

SECURITY TOOLING - DAST

- Dynamic Testing
- "Harder" than static testing
- Tools exist but should be app aware
- Better for design flaws (maybe)

SECURITY TOOLING - IAST

- DAST WITH INSTRUMENTATION
- HOOKS RUNTIME (E.G. JVM) TO MONITOR THREATS

VULNERABILITIES - HIGH

- ~~Mobile Verification Code is Susceptible to Brute-Force Attacks~~
- ~~Possible To View Other User's Messages~~
- ~~SSL Validation Disabled~~
- ~~Directory Traversal In Web Service~~

VULNERABILITIES - MEDIUM

- ~~Backups Allowed~~
- ~~No Permissions on Android IPC~~
- Sensitive Information Stored In Application Sandbox
- ~~SQL Injection in Android Content Provider~~
- ~~Weak SSL/TLS Ciphers Supported~~
- Weak Authentication in Applications

VULNERABILITIES - LOW

- ~~Apple Transport Security weakened~~
- Excessive Logging
- Lack of Anti-Debugging
- ~~Version Banner Disclosure~~
- Lack of root detection

WHAT ABOUT INFRASTRUCTURE?

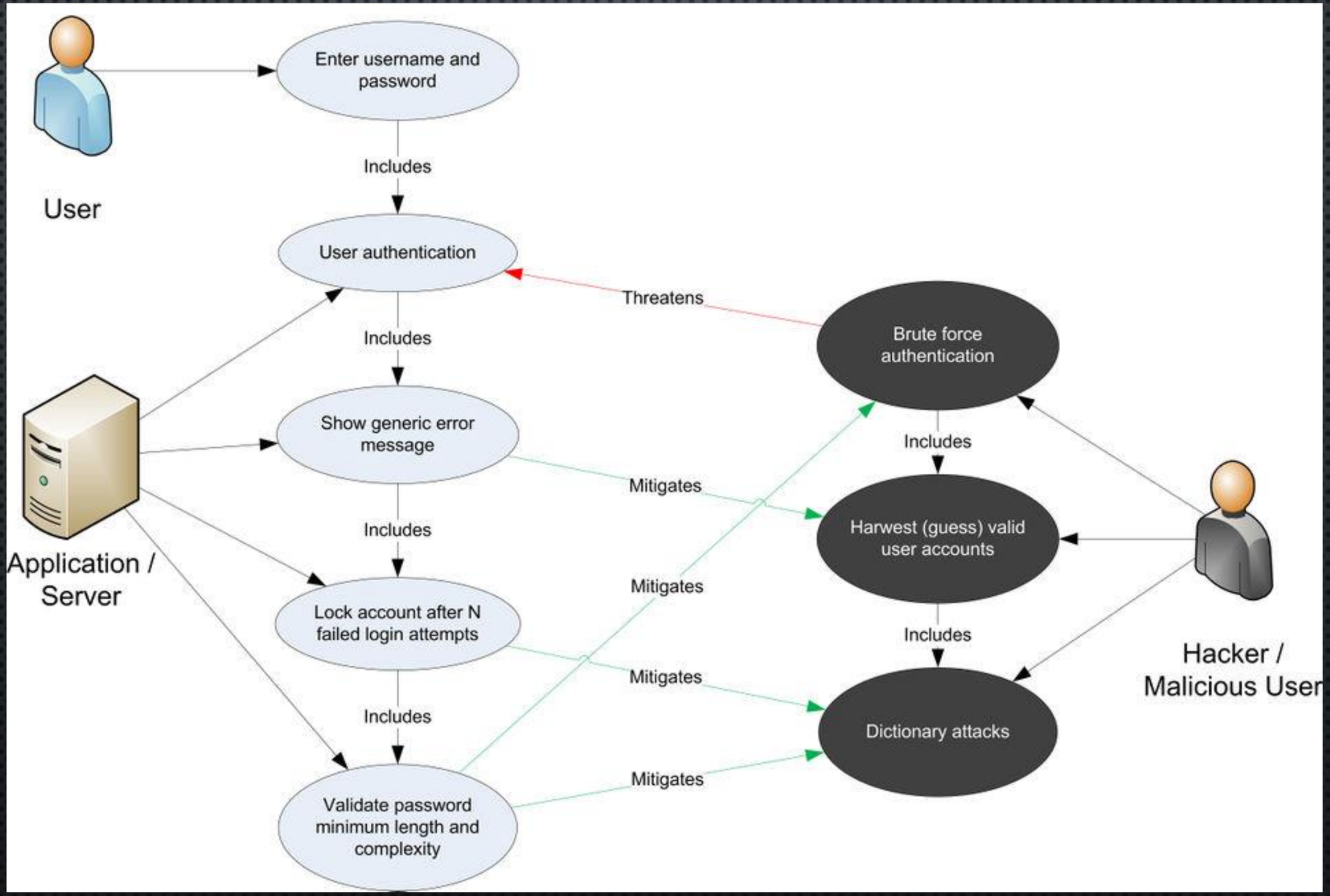
Infrastructure as code

- Key to devops
- Won't talk about it in too much detail – this is a devops conference

QA TESTING

QA TESTING

- Automation testing. How many try to automate security tests?
  - "Security isn't our job"


Enter username and password
Includes
User
User authentication
Threatens
Brute force authentication
Includes
Includes
Show generic error message
Mitigates
Harwest (guess) valid user accounts
Includes
Application / Server
Includes
Mitigates
Lock account after N failed login attempts
Includes
Hacker / Malicious User
Mitigates
Dictionary attacks
Includes
Mitigates
Validate password minimum length and complexity

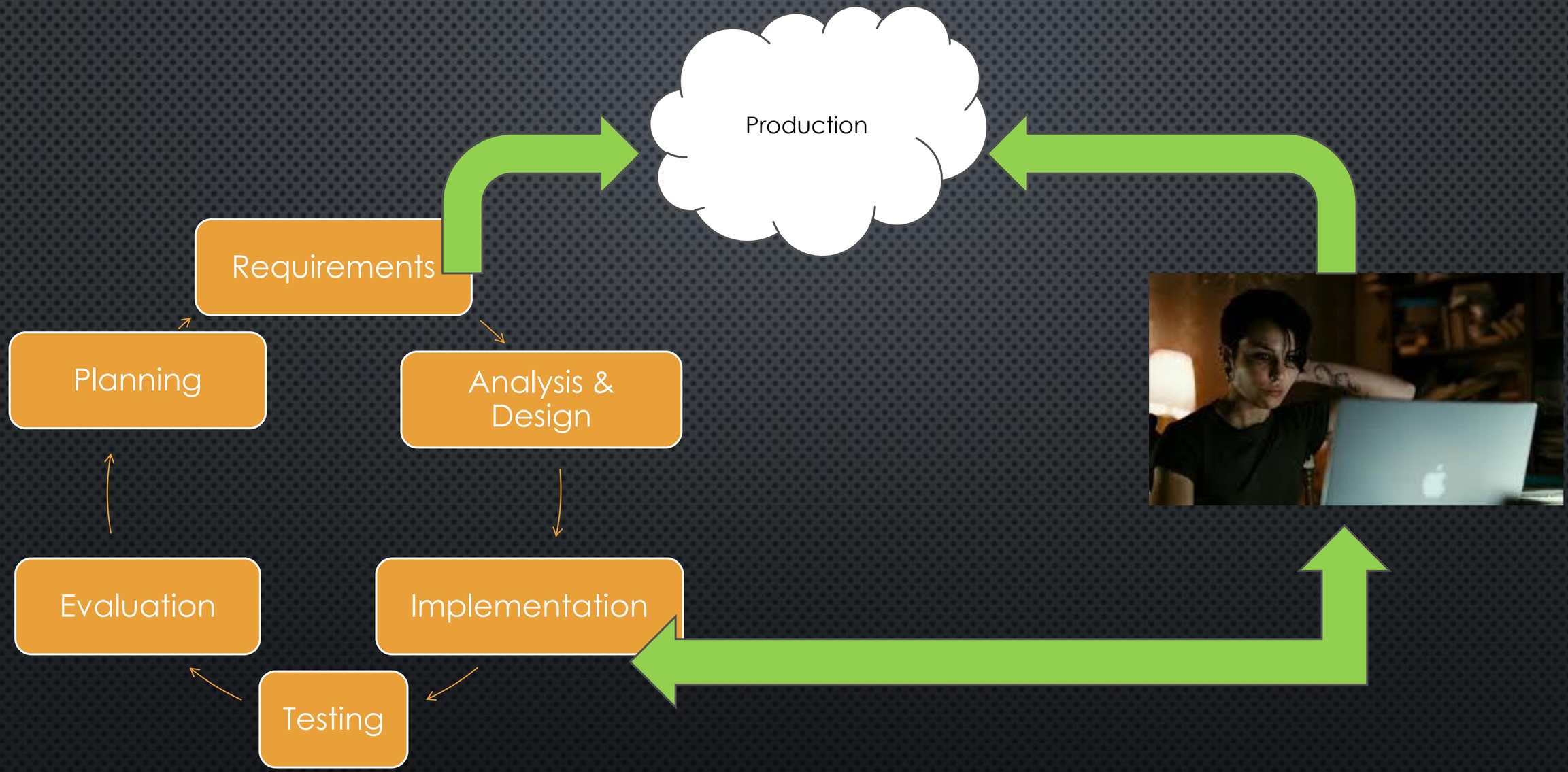
PEN TESTING

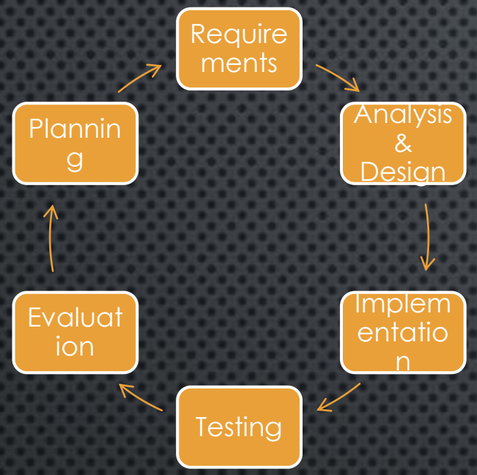
VULNERABILITIES

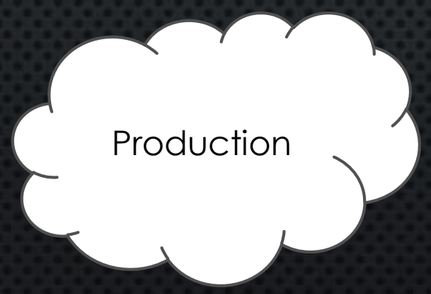
- Sensitive Information Stored In Application Sandbox
- Weak Authentication in Applications
- Excessive Logging
- Lack of Anti-Debugging
- Lack of root detection

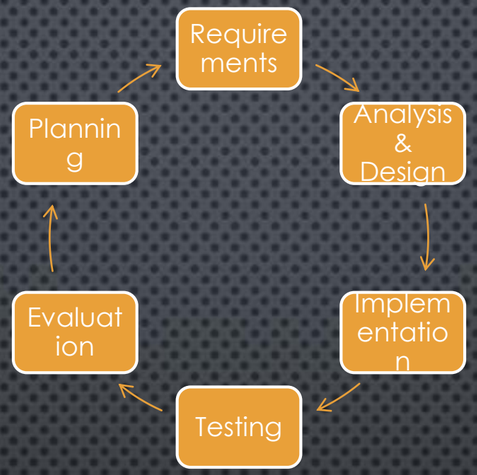
PEN TESTING

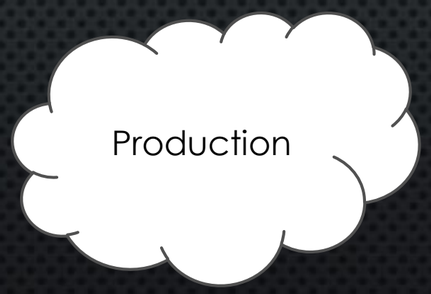
- I THOUGHT WE WERE AUTOMATING SECURITY!
- 70%* OF VULNS CAN BE FOUND BEFORE THE PEN TEST
- CHECK LIST TESTS ARE NOW OKAY!
- *YOU* DECIDE WHAT GETS TESTING SINCE EVERYTHING IS AT LEAST PARTIALLY SECURE
- (BUT PEN TESTS ARE PROBABLY STILL NEEDED FOR SOME APPLICATIONS)

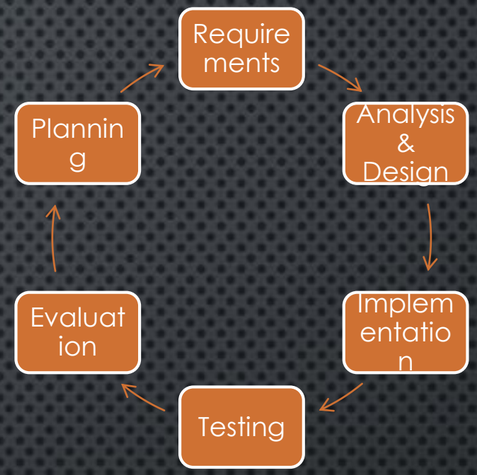

Production
Requirements
Planning
Analysis & Design
Evaluation
Implementation
Testing


Require ments
Analysis & Design
Implem entatio n
Testing
Evaluat ion
Plannin g


Production


Require ments
Analysis & Design
Implem entatio n
Testing
Evaluat ion
Plannin g


Production


Require ments
Analysis & Design
Implem entatio n
Testing
Evaluat ion
Plannin g


Production

CAPTURING RESULTS

DAST

QA Testing

SAST

Red Teaming

Vulnerabilities / Risk

QUESTIONS?

@JayHarris_Sec