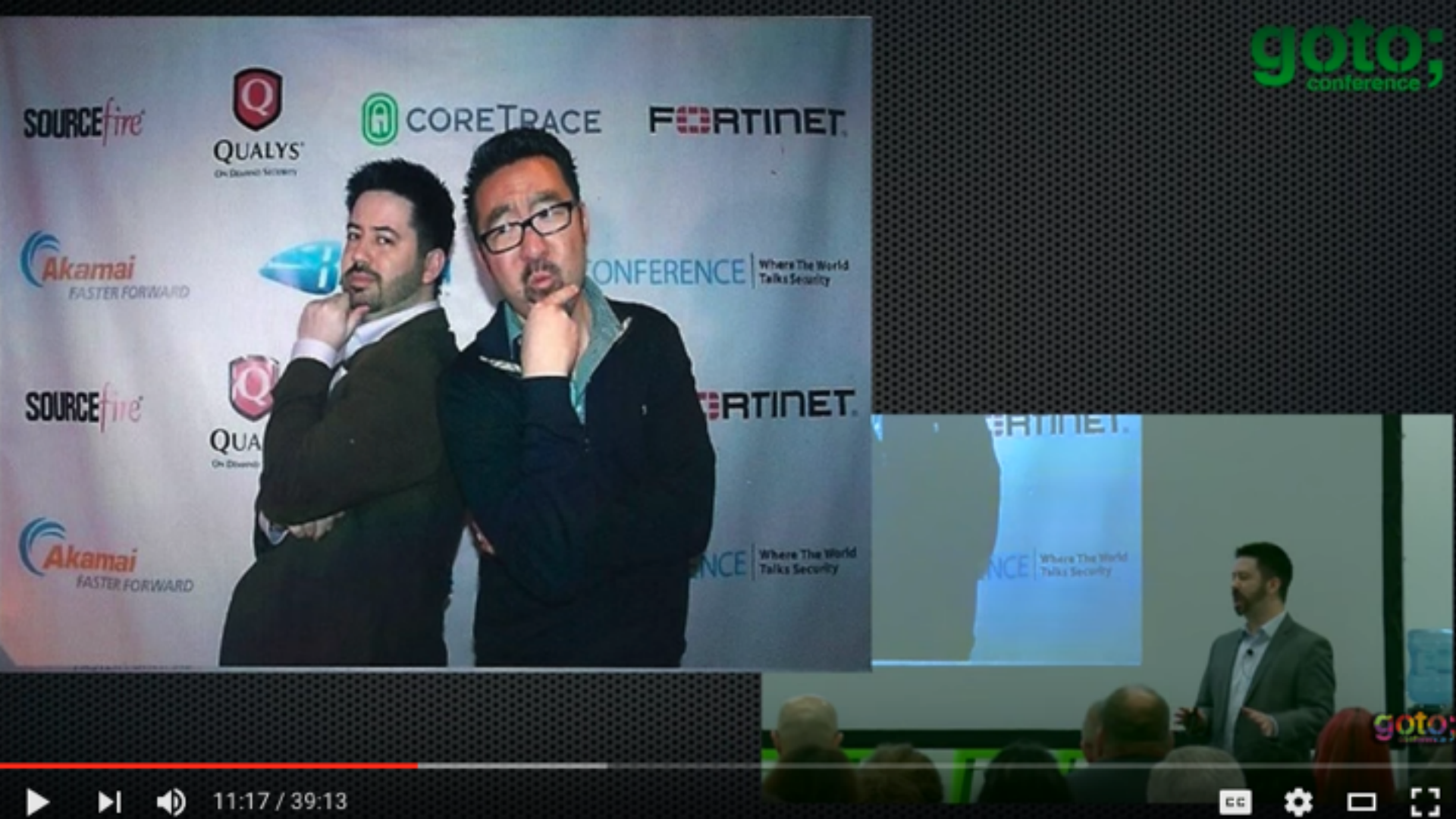
CHICAGO
INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE 2016

goto; conference

What is 'Rugged' all about?

Matt Konda


goto; conference
SOURCEfire
QUALYS
CORETRACE
FORTINET
Akamai
FASTER FORWARD
Where The World Talks Security
11:17 / 39:13


A Soldier
FOR CYBER SAFETY

I am The Cavalry

He would want me to tell you

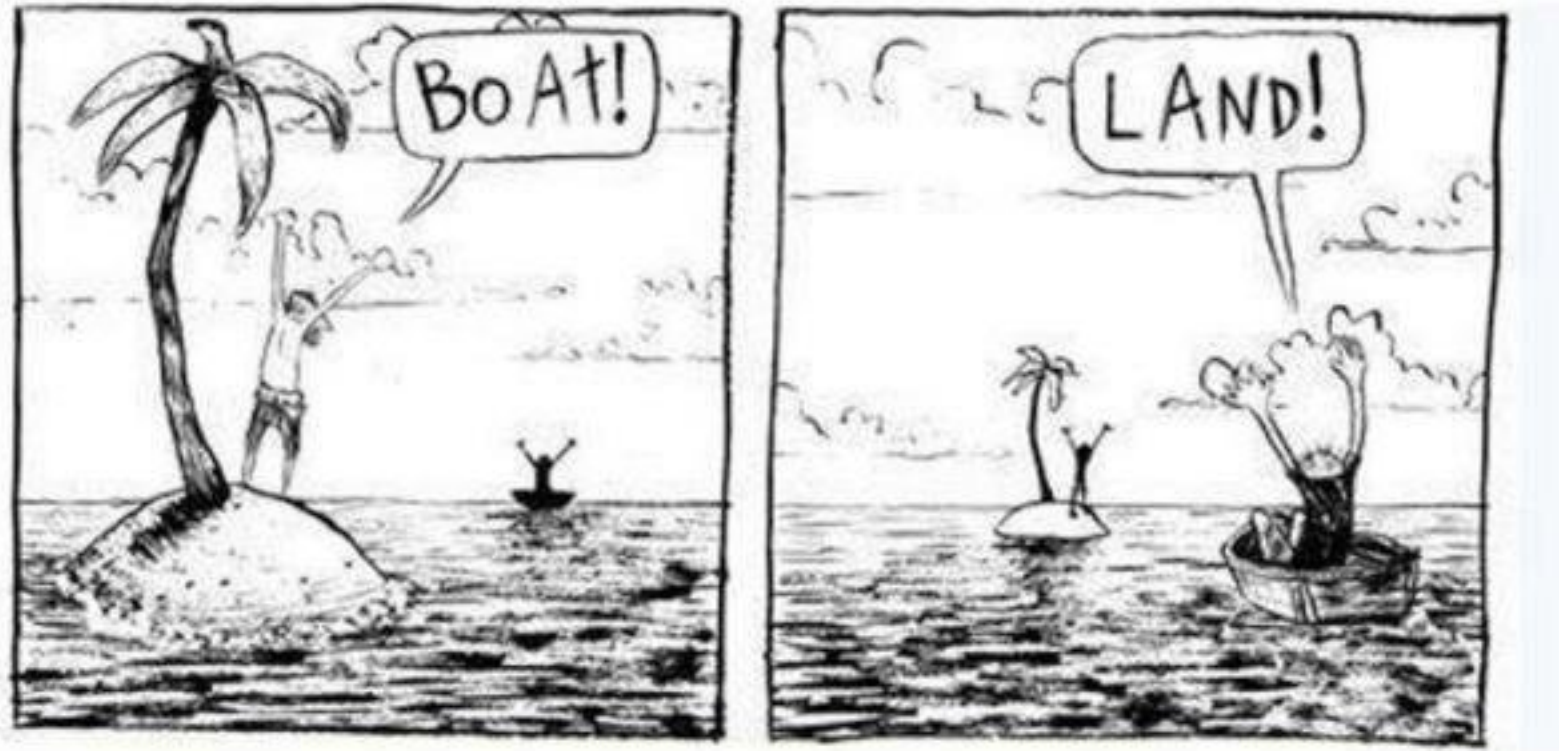
- Software is eating the world.
- DevOps and Security is a rare opportunity.
  - Makes security positive, cultural+
- Show the Rugged Manifesto
- Honey Badger = Security + DevOps …
- Empathy, Empathy, Empathy
- Bridge communities!

He would want me to emphasize

- Instrumentation
- Be Mean To Your Code
- Complexity is the Enemy
- Change Management (Automation through tooling)
- Empathy (Did I say that yet?)

He would want me to mention

- By updating our software (and it's dependencies) we can address a huge amount of attack surface.
- DevOps should be good at this.
- Empathy (Did I say that yet?)

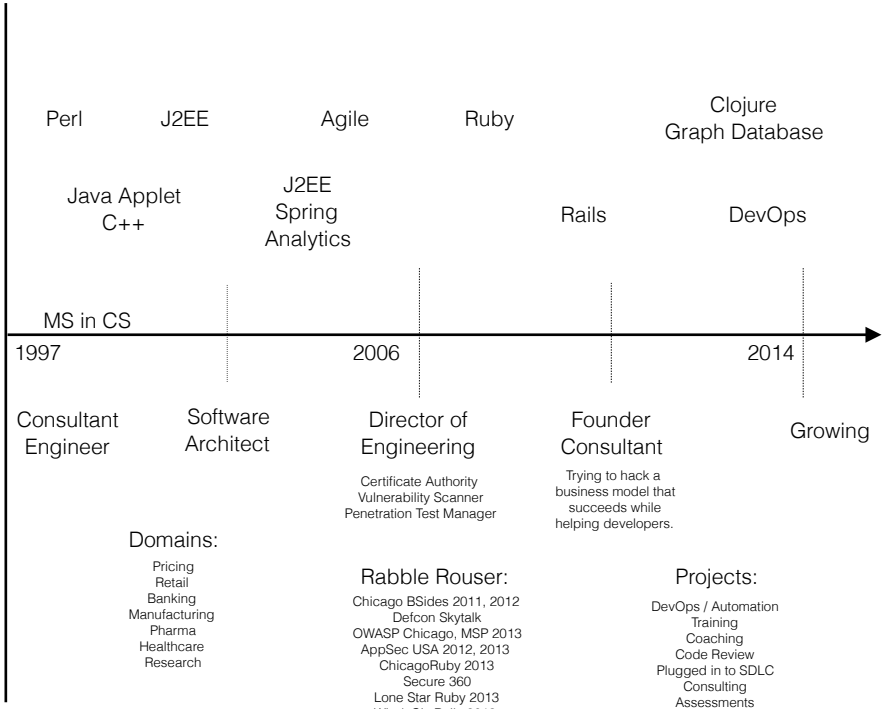


Perspective...

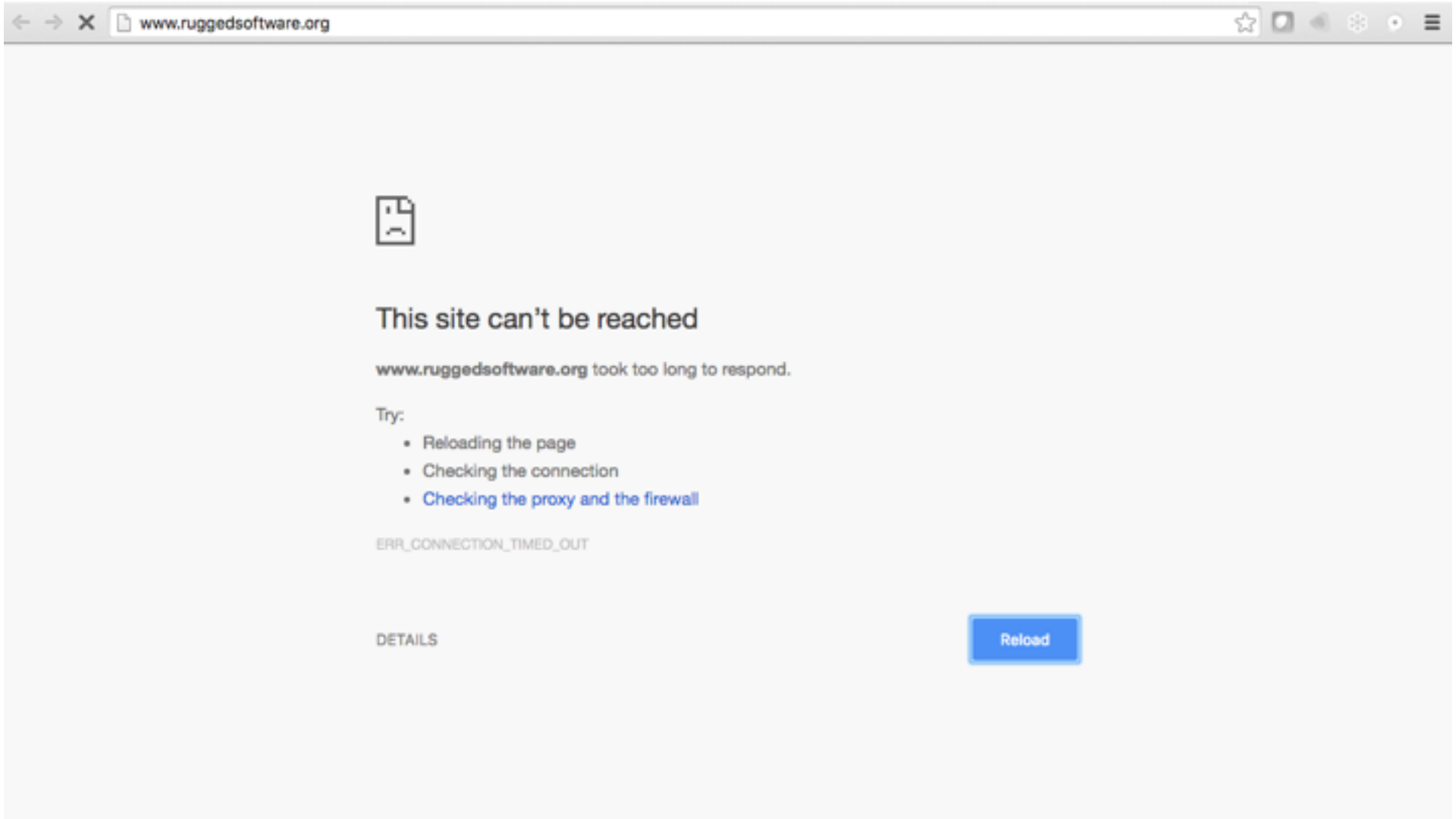
OWASP?

Introduction

@mkonda
mkonda@jemurai.com



This was a setup. Chicago style.


www.ruggedsoftware.org
This site can't be reached
www.ruggedsoftware.org took too long to respond.
Try:
Reloading the page
Checking the connection
Checking the proxy and the firewall
ERR_CONNECTION_TIMED_OUT
DETAILS
Reload

https://ruggeddev.org

Oh no! This blog's domain ruggeddev.org expired!

Unfortunately you cannot access this blog from **ruggeddev.org** any more. This domain name expired and will soon be canceled.

Please contact the owners of this website and remind them to renew this domain before it's too late. They need your help!

But in Chicago, we make the best of every situation.

Rubular

a Ruby regular expression editor

Your regular expression:

/ ^(?=.*Dev)(?=.*Sec)(?=.*Ops)(?!.*Ops.*Dev).*$ /

Your test string:

```
DevSecOps
DevOpsSec
SecDevOps
SecOpsDev
OpsDevSec
OpsSecDev
Dev
Ops
Sec
Rugged DevOps
Positive Security
DevOps
```

Match result:

DevSecOps
DevOpsSec
SecDevOps
SecOpsDev
OpsDevSec
OpsSecDev
Dev
Ops
Sec
Rugged DevOps
Positive Security
DevOps

Wrap words ☑ Show invisibles ☐ Ruby version 2.1.5

make permalink clear fields

CHICAGO
INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE 2016

goto; conference

Positive Software Security

Matt Konda

Let’s learn what we can from Rugged (applied to DevOps)

I recognize that software has become a foundation of our modern world.

I recognize the awesome responsibility that comes with this foundational role.

I recognize that my code will be used in ways I cannot anticipate, in ways it was not designed, and for longer than it was ever intended.

I recognize that my code will be attacked by talented and persistent adversaries who threaten our physical, economic and national security.

I recognize these things – and I choose to be rugged.

I recognize that software has become a foundation of our modern world.

I recognize the awesome responsibility that comes with this foundational role.

I recognize that my code will be used in ways I cannot anticipate, in ways it was not designed, and for longer than it was ever intended.

I recognize that my code will be attacked by talented and persistent adversaries who threaten our physical, economic and national security.

I recognize these things – and I choose to be rugged.

I recognize that software has become a foundation of our modern world.

I recognize the awesome responsibility that comes with this foundational role.

I recognize that my code will be used in ways I cannot anticipate, in ways it was not designed, and for longer than it was ever intended.

I recognize that my code will be attacked by talented and persistent adversaries who threaten our physical, economic and national security.

I recognize these things – and I choose to be rugged.

Reminiscent of the Agile Manifesto Perhaps?

Let’s talk about adversaries…


MR. ROBO

This year, organized crime became the most frequently seen threat actor for Web App Attacks.

Source: Verizon 2015 Data Breach Investigations Report


iMac

Threat model

Security Examples

```
SELECT "orders".* FROM "orders"
WHERE
  (rewards_code = 'a')
union select id, 'product', 1, 1,
  'cc', 'cvv', 'expiration',
email as first_name,
encrypted_password as last_name,
created_at, updated_at,
   id, 'reward' from users; --')
```

Getting Rugged?

Train.

Search for string concatenation: +, append prefer parameterized queries!

Do code review.

Use static analysis.

Use web app scanning.

Output Encoding

<	&lt;
>	&gt;

Getting Rugged?

Train.

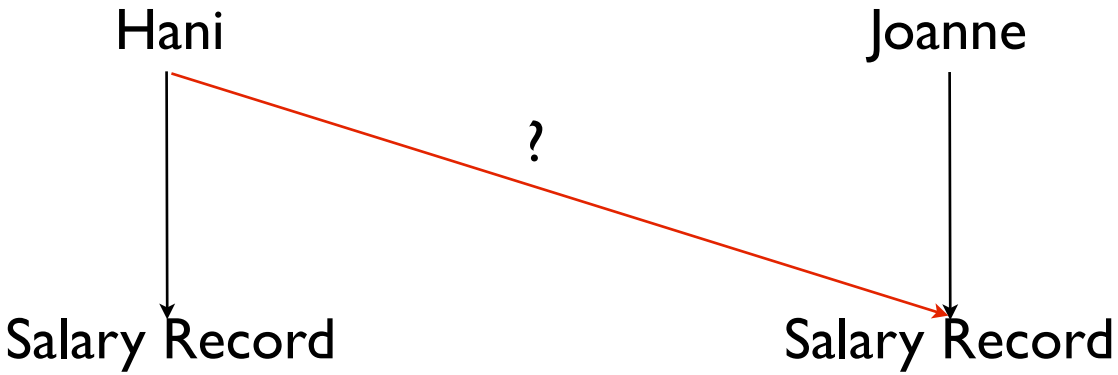
Search for {{{, innerHTML, .raw, utext, etc.

Do code review.

Use static analysis.

Use web app scanning.

Insecure Direct Object Reference



Authorization fail!

```
Given(/^a new project created by a user$/) do
  uuid = SecureRandom.uuid
  @user1 = "fb_user_1_#{uuid}@jemurai.com"
  register_as_user(@user1, "password")
#  logout(@user1)
#  login_as_user(@user1, 'password')
  new_project("Insecure Deirect Object Reference #{uuid}", "Forceful Browsing Desc")
  @url = current_url
end

When(/^a different person attempts to access the project$/) do
  logout(@user1)
  uuid = SecureRandom.uuid
  @user2 = "fb_user_2_#{uuid}@jemurai.com"
  register_as_user(@user2, "password")
#  logout(@user2)
#  login_as_user(@user2, 'password')
end

Then(/^the system should prevent access$/) do
  visit @url
  expect(page).not_to have_content "Forceful Browsing Desc"
end
```

The Sydney Morning Herald
BusinessDay

Menulog customers access other people's private details and orders

May 19, 2016 - 4:30PM

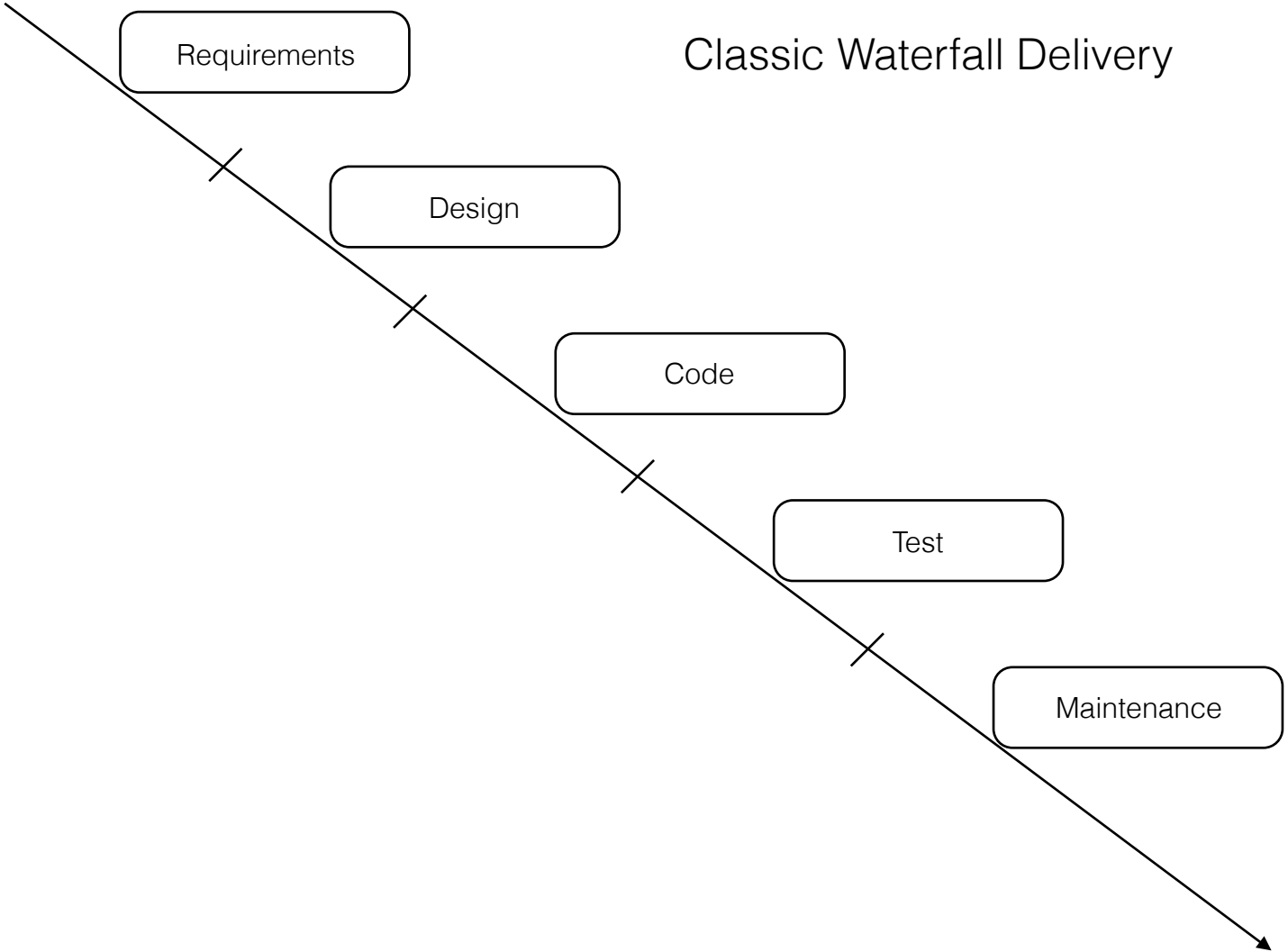
Esther Han
Consumer Affairs Editor
View more articles from Esther Han
Follow Esther on Twitter Follow Esther on Google+ Email Esther

Some Specifics Around Process

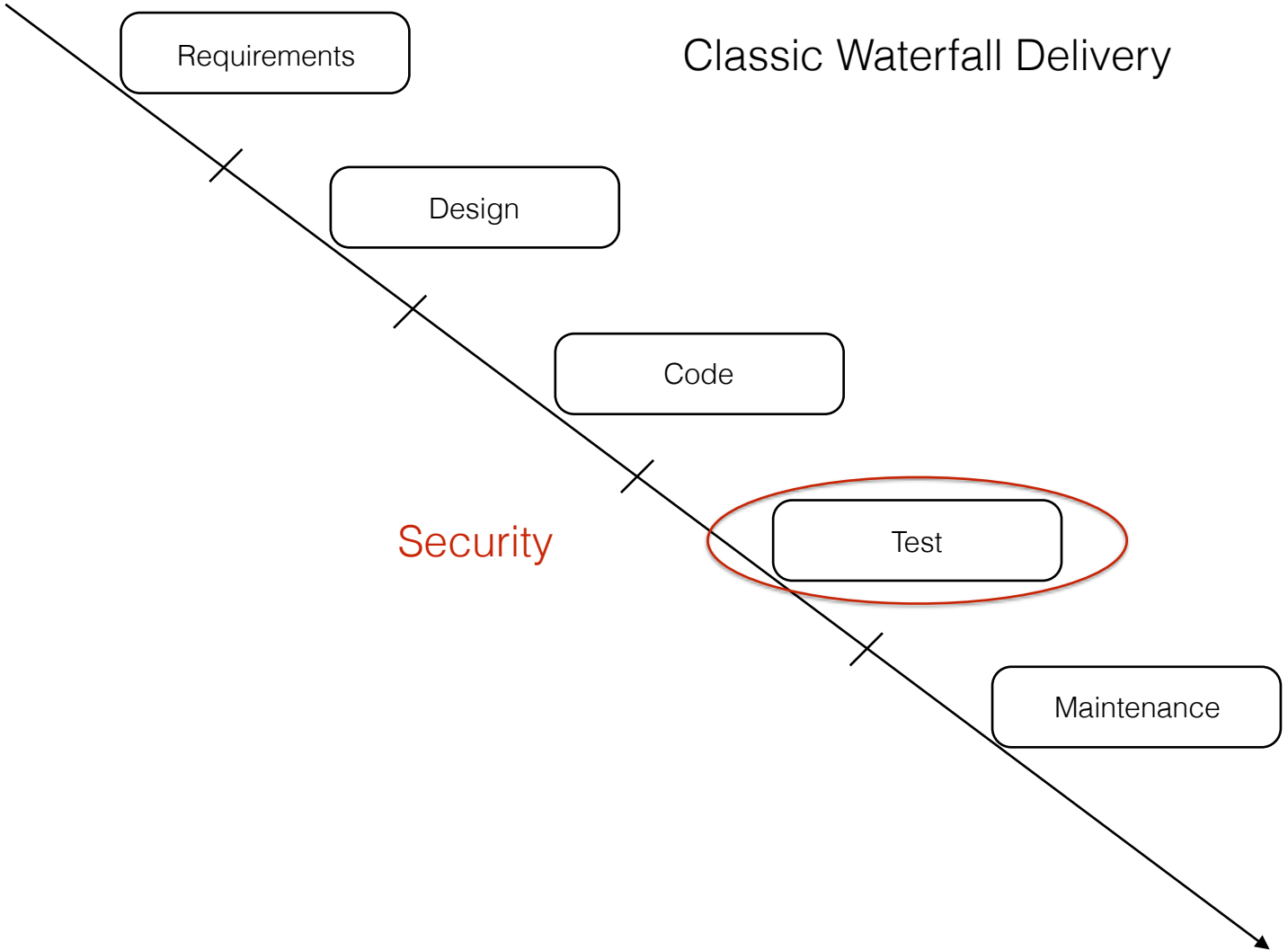
Security in the SDLC

- Building software is a process.
- The best way to make software secure is to make security part of the process.
- There are many ways to do this - none is perfect.
- Find a way to make the security fit your process.

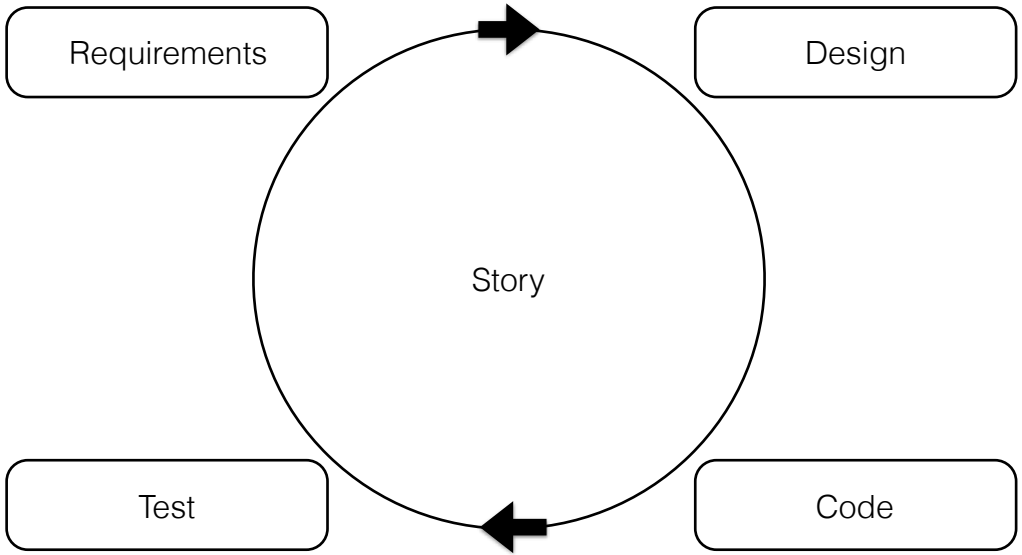
Classic Waterfall Delivery



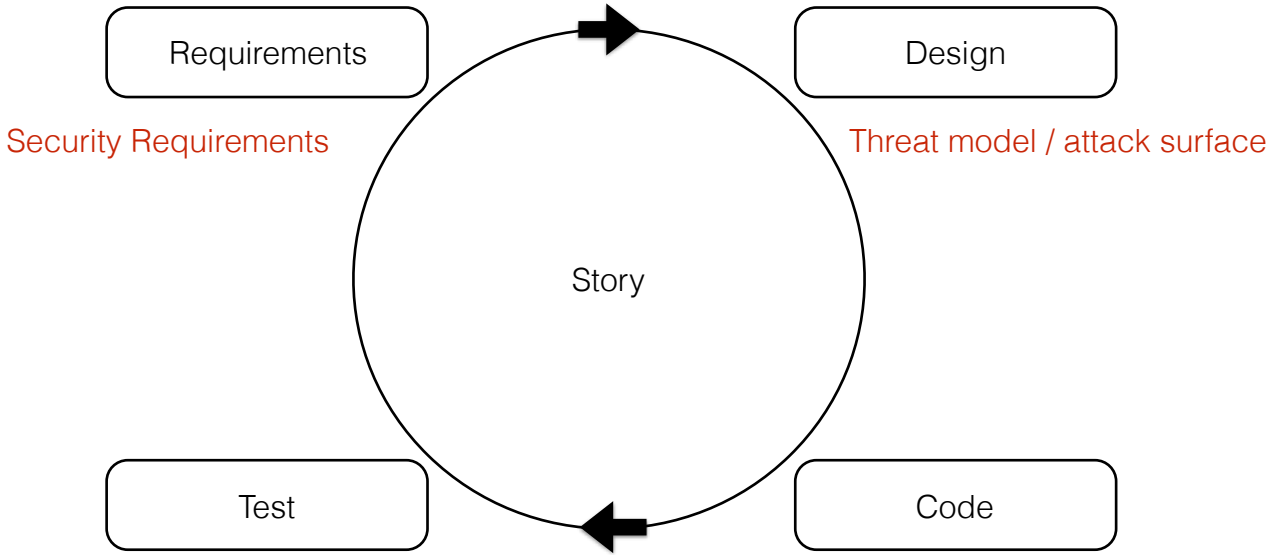
Classic Waterfall Delivery



Continuous Delivery: The Unit of Work is a Story



Continuous Delivery: The Unit of Work is a Story



Security Requirements

Threat model / attack surface

Security Unit Tests

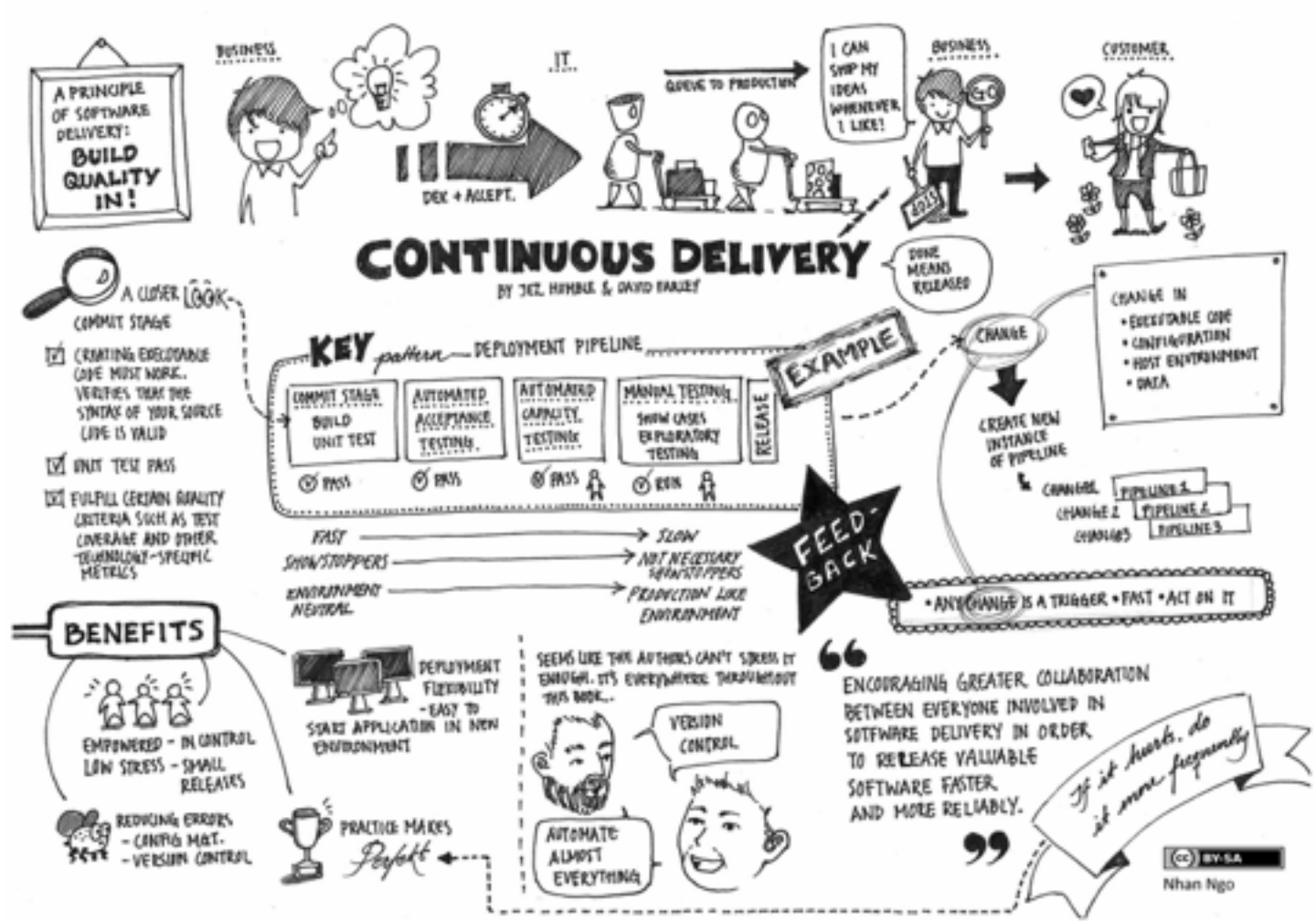
Exploratory Testing

Static Analysis on Commit

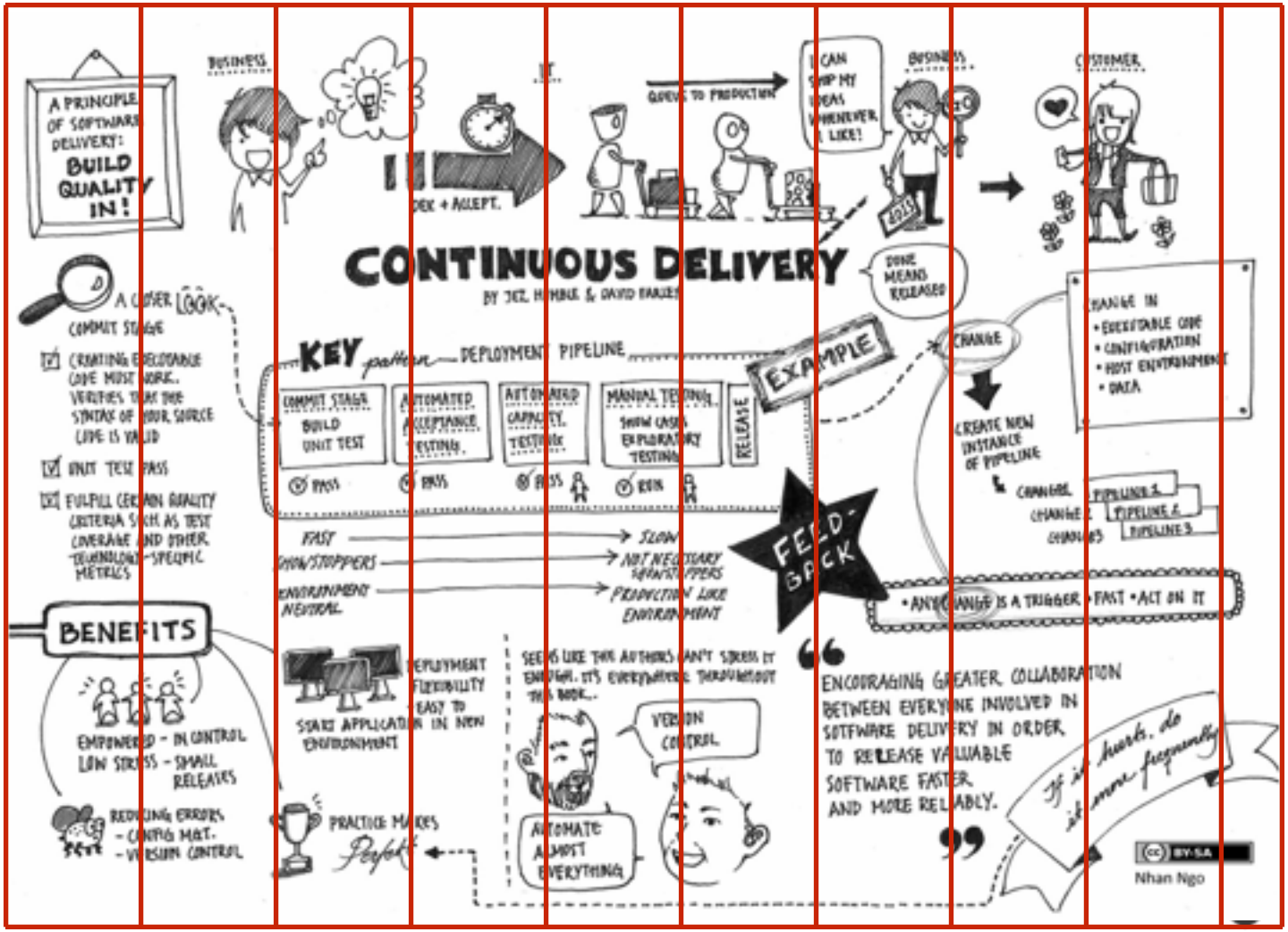
Code Review

Checklists

Understand Dependencies

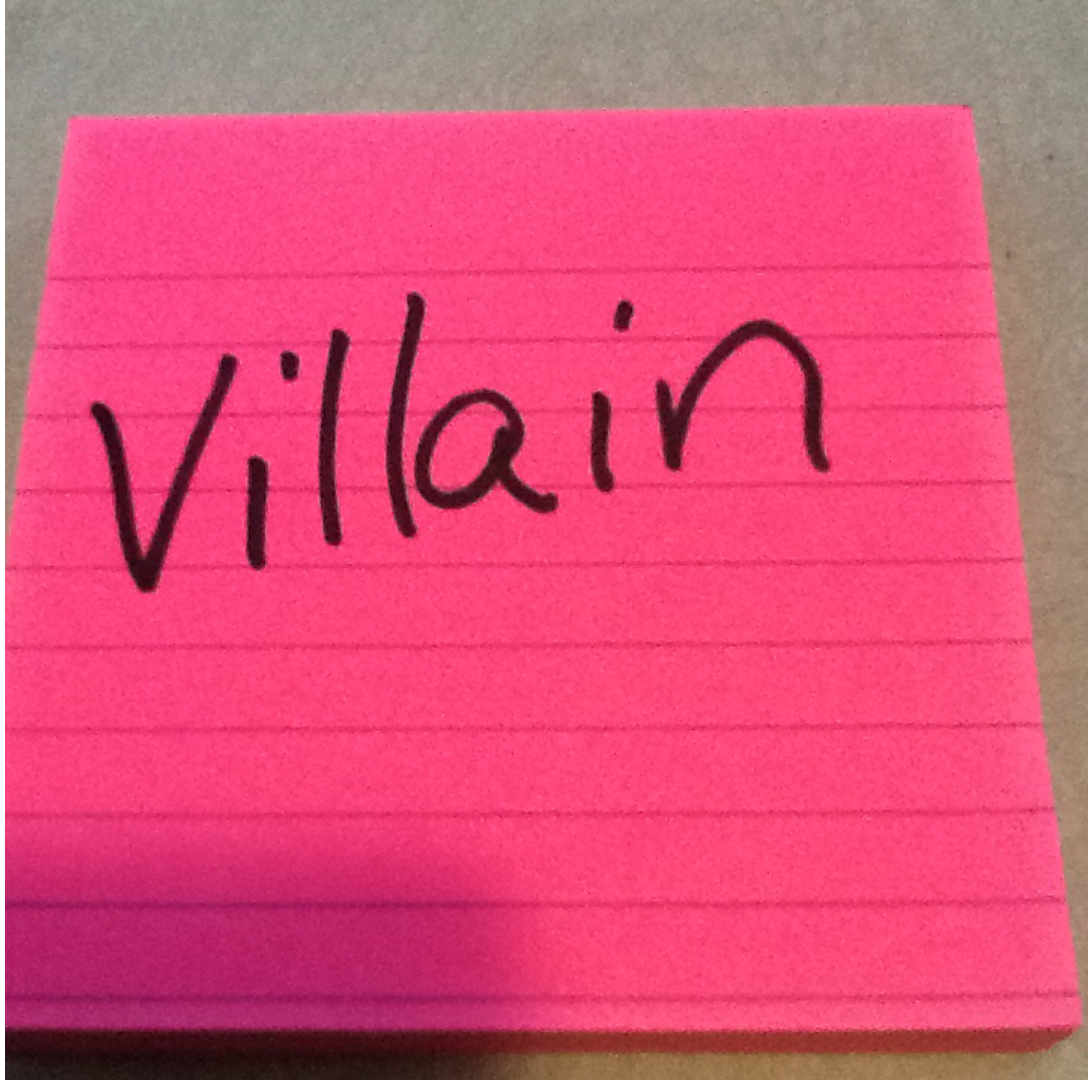

A PRINCIPLE OF SOFTWARE DELIVERY: BUILD QUALITY IN!
BUSINESS
IT
DEV + ALLEPT.
QUEUE TO PRODUCTION
I CAN SHIP MY IDEAS WHENEVER I LIKE!
BUSINESS
GO
CUSTOMER
CONTINUOUS DELIVERY
BY JEZ HUMBLE & DAVID FARLEY
DONE MEANS RELEASED
A CLOSER LOOK
COMMIT STAGE
CREATING EXECUTABLE CODE MUST WORK. VERIFIES THAT THE SYNTAX OF YOUR SOURCE CODE IS VALID
UNIT TEST PASS
FULFILL CERTAIN QUALITY CRITERIA SUCH AS TEST COVERAGE AND OTHER TECHNOLOGY-SPECIFIC METRICS
KEY pattern — DEPLOYMENT PIPELINE
EXAMPLE
COMMIT STAGE
BUILD
UNIT TEST
AUTOMATED ACCEPTANCE TESTING
AUTOMATED CAPACITY TESTING
MANUAL TESTING
SHOW CASES
EXPLORATORY TESTING
RELEASE
PASS
PASS
PASS
RUN
FAST
SHOW-STOPPERS
ENVIRONMENT NEUTRAL
SLOW
NOT NECESSARY SHOW-STOPPERS
PRODUCTION LIKE ENVIRONMENT
FEED-BACK
CHANGE
CHANGE IN
• EXECUTABLE CODE
• CONFIGURATION
• HOST ENVIRONMENT
• DATA
CREATE NEW INSTANCE OF PIPELINE
CHANGE
CHANGE 2
CHANGE 3
PIPELINE 1
PIPELINE 2
PIPELINE 3
• ANY CHANGE IS A TRIGGER • FAST • ACT ON IT
BENEFITS
EMPOWERED - IN CONTROL
LOW STRESS - SMALL RELEASES
REDUCING ERRORS
- CONFIG MGT.
- VERSION CONTROL
DEPLOYMENT FLEXIBILITY
- EASY TO START APPLICATION IN NEW ENVIRONMENT
PRACTICE MAKES Perfekt
SEEMS LIKE THIS AUTHORS CAN'T STRESS IT ENOUGH. IT'S EVERYWHERE THROUGHOUT THIS BOOK..
VERSION CONTROL
AUTOMATE ALMOST EVERYTHING
“ENCOURAGING GREATER COLLABORATION BETWEEN EVERYONE INVOLVED IN SOFTWARE DELIVERY IN ORDER TO RELEASE VALUABLE SOFTWARE FASTER AND MORE RELIABLY.”
If it hurts, do it more frequently
BY-SA
Nhan Ngo

Classic security sees this and wants to …


A PRINCIPLE OF SOFTWARE DELIVERY: BUILD QUALITY IN!
BUSINESS
IT
QUICK TO PRODUCTION
I CAN SHIP MY IDEAS WHENEVER I LIKE!
BUSINESS
CUSTOMER
CONTINUOUS DELIVERY
BY JEZ HUMBLE & DAVID FARLEY
DONE MEANS RELEASED
A CLOSER LOOK
COMMIT STAGE
CREATING EXECUTABLE CODE MUST WORK. VERIFIES THAT THE SYNTAX OF YOUR SOURCE CODE IS VALID
UNIT TEST PASS
FULFILL CERTAIN QUALITY CRITERIA SUCH AS TEST COVERAGE AND OTHER TECHNOLOGY-SPECIFIC METRICS
KEY pattern — DEPLOYMENT PIPELINE
EXAMPLE
COMMIT STAGE
BUILD UNIT TEST
AUTOMATED ACCEPTANCE TESTING
AUTOMATED CAPACITY TESTING
MANUAL TESTING
SHOW CASES EXPLORATORY TESTING
RELEASE
PASS
PASS
PASS
RUN
CHANGE
CHANGE IN
• EXECUTABLE CODE
• CONFIGURATION
• HOST ENVIRONMENT
• DATA
CREATE NEW INSTANCE OF PIPELINE
CHANGE
CHANGE
CHANGE
PIPELINE 3
PIPELINE 4
PIPELINE 5
FAST
SHOW-STOPPERS
ENVIRONMENT NEUTRAL
SLOW
NOT NECESSARY SHOW-STOPPERS
PRODUCTION LIKE ENVIRONMENT
FEED-BACK
• ANY CHANGE IS A TRIGGER • FAST • ACT ON IT
BENEFITS
EMPOWERED - IN CONTROL
LOW STRESS - SMALL RELEASES
REDUCING ERRORS
- CONFIG MGT.
- VERSION CONTROL
DEPLOYMENT FLEXIBILITY
EASY TO START APPLICATION IN NEW ENVIRONMENT
PRACTICE MAKES Perfect
SEEMS LIKE THIS AUTHORS CAN'T STRESS IT ENOUGH. IT'S EVERYWHERE THROUGHOUT THIS BOOK..
VERSION CONTROL
AUTOMATE ALMOST EVERYTHING
"ENCOURAGING GREATER COLLABORATION BETWEEN EVERYONE INVOLVED IN SOFTWARE DELIVERY IN ORDER TO RELEASE VALUABLE SOFTWARE FASTER AND MORE RELIABLY."
If it hurts, do it more frequently
BY-SA
Nhan Ngo

Baseline Security Requirements

ARE STAKEHOLDERS ASKING FOR SECURITY?


Villain

Story Points

Estimates to Include Security Considerations

Here’s why.

Agile metrics

Credit: rallydev.com

Story Review

Incremental Code Review

Continuous Integration

Back to Dashboard
Status
Changes
Workspace
Build Now
Delete Project
Configure
GitHub

Build History trend

find

#97	May 24, 2016 2:33 PM	
#96	May 24, 2016 2:31 PM	
#95	May 24, 2016 2:21 PM	
#94	May 19, 2016 9:16 PM	
#93	May 19, 2016 9:14 PM	
#92	May 19, 2016 9:14 PM	
#91	May 19, 2016 7:46 PM	
#90	May 19, 2016 7:31 PM	
#89	May 19, 2016 7:16 PM	
#88	May 19, 2016 7:01 PM	
#87	May 19, 2016 6:46 PM	

Project TriageBuild

Triage Build

Workspace

Recent Changes

Permalinks

- Last build (#96), 2 min 0 sec ago
- Last stable build (#11), 4 mo 23 days ago
- Last successful build (#11), 4 mo 23 days ago
- Last failed build (#96), 2 min 0 sec ago
- Last unsuccessful build (#96), 2 min 0 sec ago
- Last completed build (#96), 2 min 0 sec ago

Back to Project
Status
Changes
Console Output
View as plain text
Edit Build Information
Git Build Data
No Tags
Previous Build

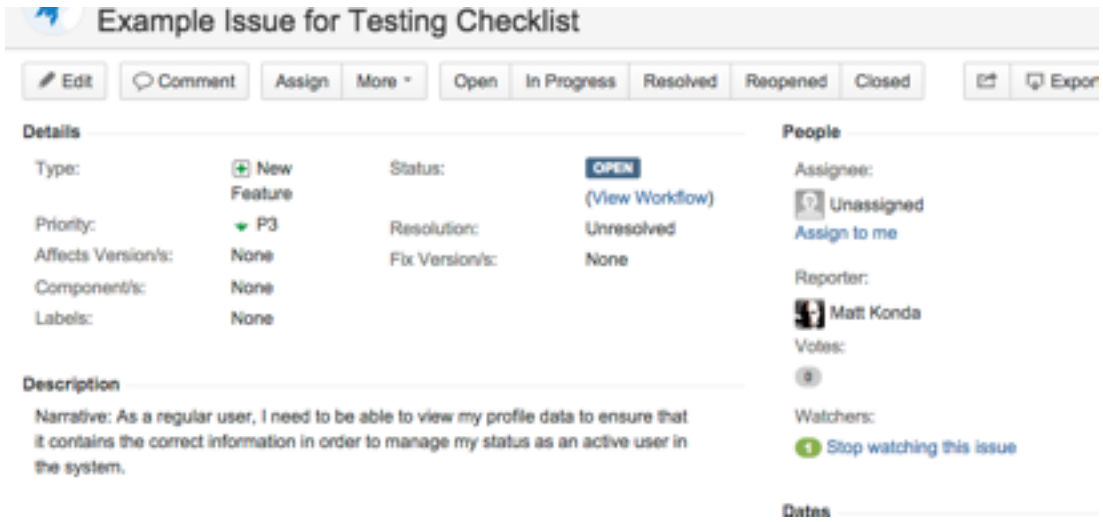
Console Output

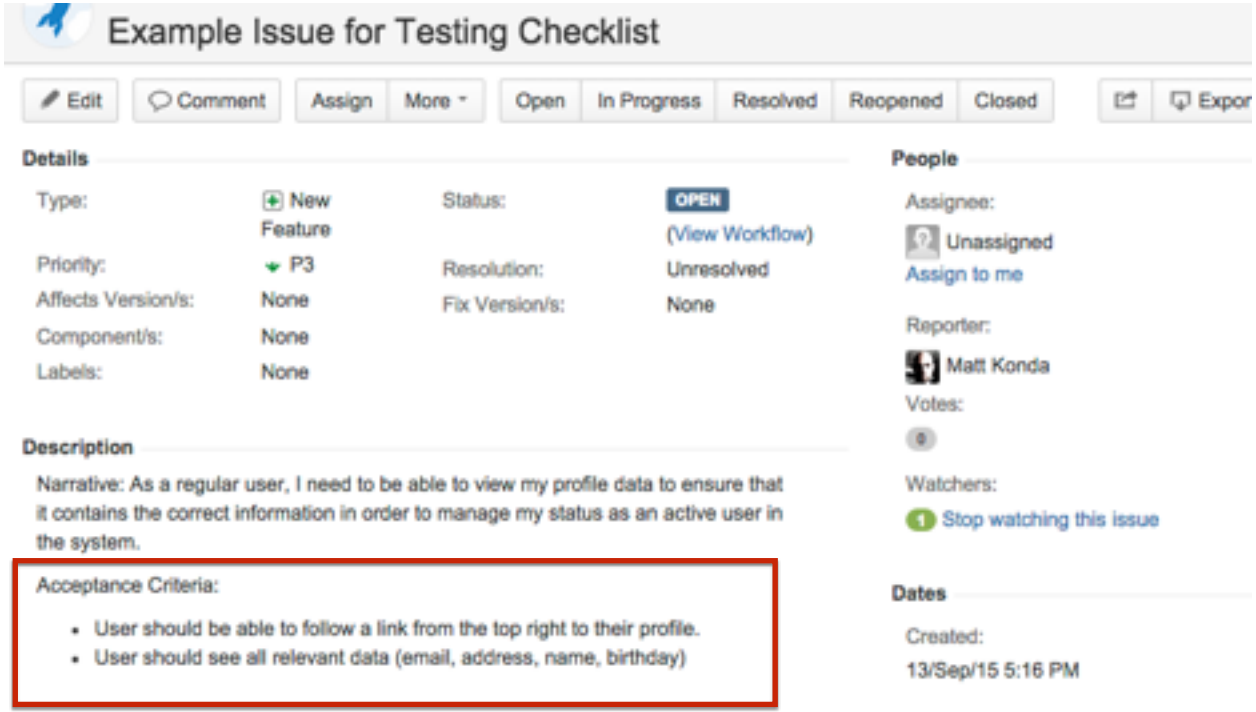
```
Started by user anonymous
Building in workspace /Users/mk/.jenkins/workspace/TriageBuild
 > git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/Jemurai/triage.git # timeout=10
Fetching upstream changes from https://github.com/Jemurai/triage.git
 > git --version # timeout=10
 > git -c core.askpass=true fetch --tags --progress https://github.com/Jemurai/triage.git +refs
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
 > git rev-parse refs/remotes/origin/origin/master^{commit} # timeout=10
Checking out Revision ef295cbc4b76f96e34abb1fab3e761405a9ef52f (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f ef295cbc4b76f96e34abb1fab3e761405a9ef52f
 > git rev-list ef295cbc4b76f96e34abb1fab3e761405a9ef52f # timeout=10
executing script 'Pipeline Build Step'
[TriageBuild] $ /bin/sh /area52/tomcat/temp/build_step_template72493073143623392.sh
Starting Pipeline Tool
Script executed from: /Users/mk/.jenkins/workspace/TriageBuild
Loading scanner...
Mounting ... /tmp/20965/
Mounting target: /tmp/20965/
Checking about mounting /tmp/20965/ with #<Pipeline::DockerMounter:0x00000001ea67e8>
In Docker mounter, target: /tmp/20965/ became: /20965/ ... wondering if it matched .docker
Checking about mounting /tmp/20965/ with #<Pipeline::FileSystemMounter:0x00000001ea6658>
Mounting /tmp/20965/ with #<Pipeline::FileSystemMounter:0x00000001ea6658>
Mounted /tmp/20965/ with #<Pipeline::FileSystemMounter:0x00000001ea6658>
Processing target.../tmp/20965/
Running tasks in stage: wait
Couldn't connect to server
Running tasks in stage: mount
Couldn't connect to server
Running tasks in stage: file
Couldn't connect to server
Running tasks in stage: code
code - Brakeman - #<Set:0x00000001e0fa28>
code - BundleAudit - #<Set:0x00000001dc9fc8>
Rootpath: /tmp/20965/
Not sure how to handle line: Remote DoS
Not sure how to handle line: Vulnerabilities found!

code - ESLint - #<Set:0x00000001d64678>
ESLint/ScanJS
ESLint Config Path: /home/pipe/line/pipeline/lib/pipeline/tasks
```

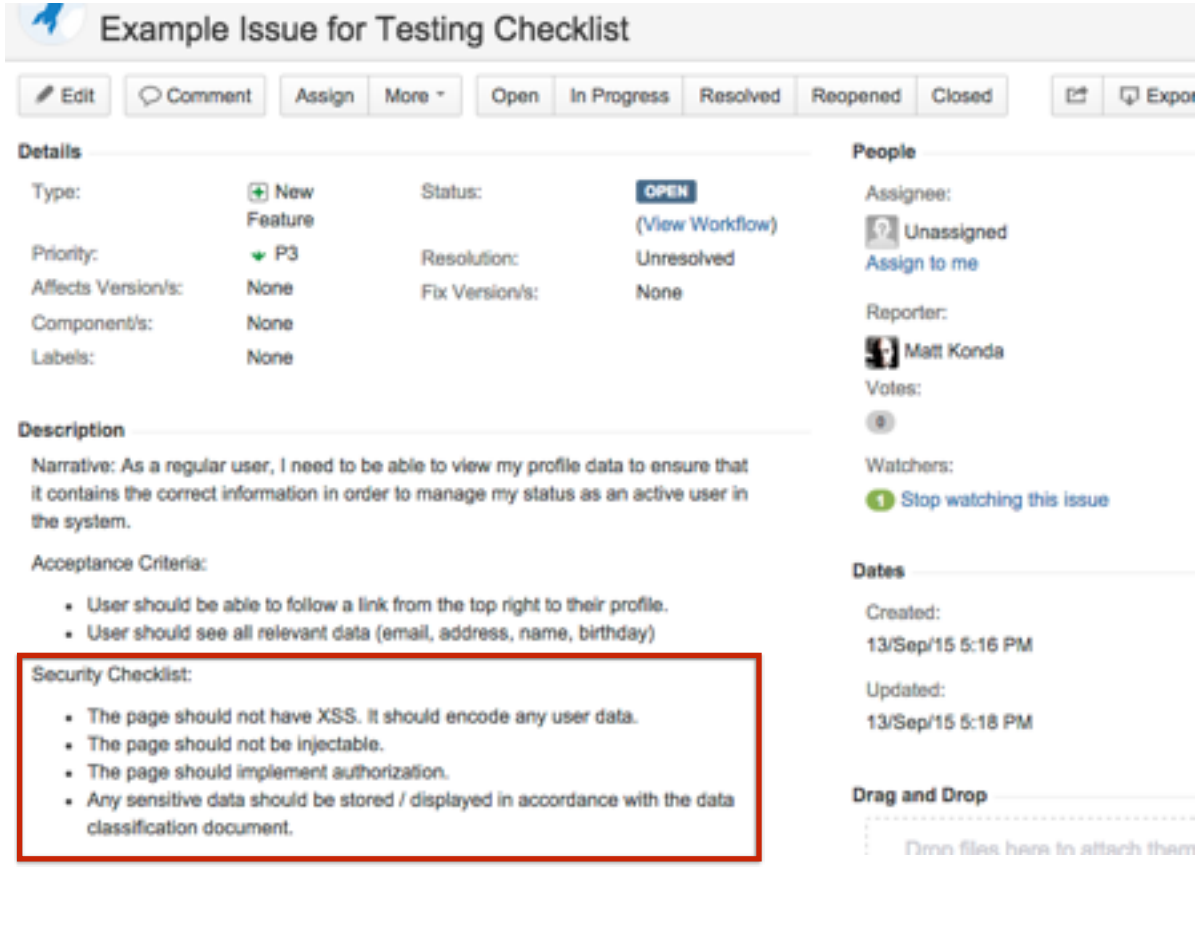
```
  allow potentially malicious code to be hidden within secure code, and
  activated by the minification process.

  For more information, consult:
  https://zyan.scripts.mit.edu/blog/backdooring-js/

Solution: upgrade to >= 2.7.2

Vulnerabilities found!


Finding: -d
        Description: Package jquery-1.7.2 has known security issues
        Timestamp: 2016-05-24 19:35:48 +0000
        Source: {:scanner=>"RetireJS", :file=>"app/assets/javascripts/jquery.js", :line=>nil, :code=>nil}
        Severity: 2
        Fingerprint:  f2b85f623df1559057eed57d93e52bde82e9ae4f2faa9a49f0c01d18e9bacc40
        Detail:  http://bugs.jquery.com/ticket/11290\nhttp://research.insecurelabs.org/jquery/test/
Finding: -d
        Description: Contains word: password
        Timestamp: 2016-05-24 19:35:48 +0000
        Source: SFL:/tmp/20965/app/views/devise/mailer/reset_password_instructions.html.erb
        Severity: unknown
        Fingerprint:  TBD
        Detail:
Finding: -d
        Description: Contains word: password
        Timestamp: 2016-05-24 19:35:48 +0000
        Source: SFL:/tmp/20965/app/views/devise/passwords
        Severity: unknown
        Fingerprint:  TBD
        Detail:
Finding: -d
        Description: Potential Ruby On Rails database configuration file
        Timestamp: 2016-05-24 19:35:48 +0000
        Source: SFL:/tmp/20965/config/database.yml
        Severity: unknown
        Fingerprint:  TBD
        Detail:  Might contain database credentials.
Finding: -d
        Description: Ruby On Rails database schema file
        Timestamp: 2016-05-24 19:35:48 +0000
        Source: SFL:/tmp/20965/db/schema.rb
        Severity: unknown
        Fingerprint:  TBD
        Detail:  Contains information on the database schema of a Ruby On Rails application.
Build step 'Execute managed script' marked build as failure
Finished: FAILURE
```

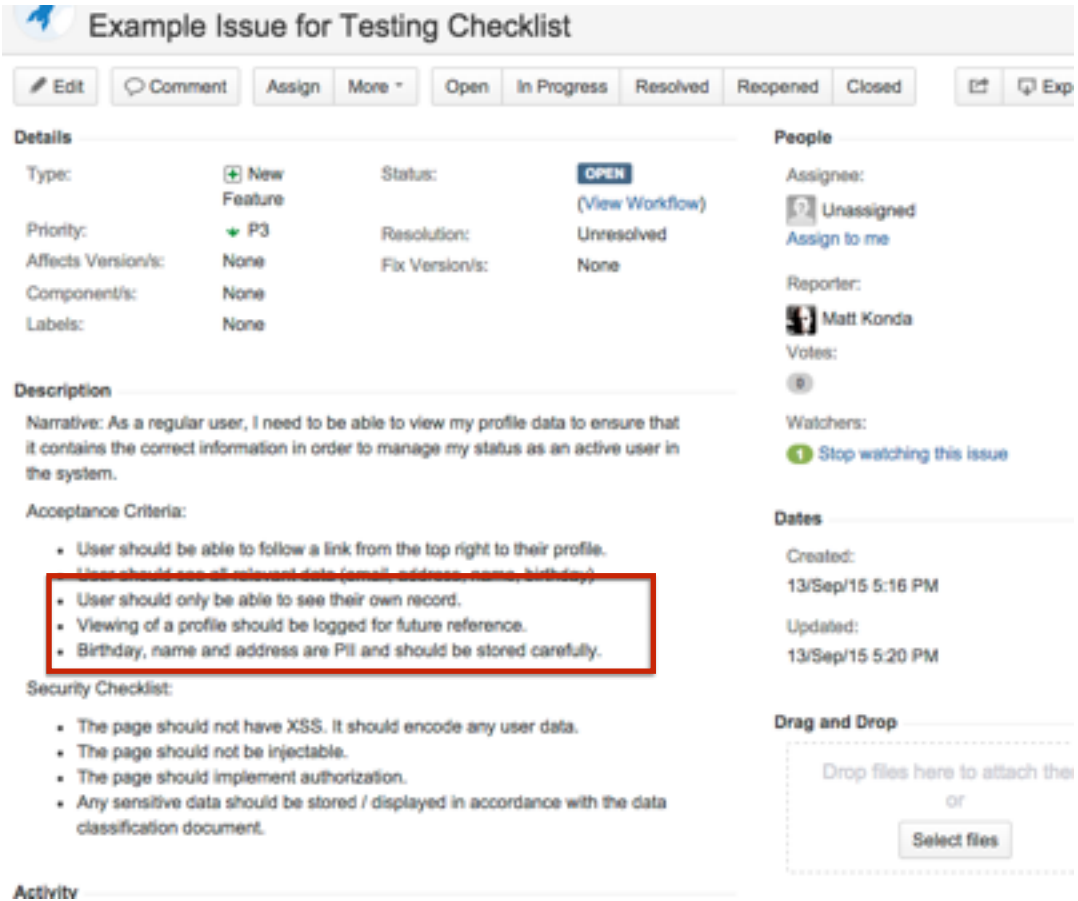
Static Analysis

Checklists


Example Issue for Testing Checklist
Edit
Comment
Assign
More
Open
In Progress
Resolved
Reopened
Closed
Details
Type:
New Feature
Priority:
P3
Affects Version/s:
None
Component/s:
None
Labels:
None
Status:
OPEN
(View Workflow)
Resolution:
Unresolved
Fix Version/s:
None
Description
Narrative: As a regular user, I need to be able to view my profile data to ensure that it contains the correct information in order to manage my status as an active user in the system.
People
Assignee:
Unassigned
Assign to me
Reporter:
Matt Konda
Votes:
0
Watchers:
1
Stop watching this issue
Dates


Example Issue for Testing Checklist
Edit
Comment
Assign
More
Open
In Progress
Resolved
Reopened
Closed
Expor
Details
Type:
New Feature
Status:
OPEN
(View Workflow)
Priority:
P3
Resolution:
Unresolved
Affects Version/s:
None
Fix Version/s:
None
Component/s:
None
Labels:
None
People
Assignee:
Unassigned
Assign to me
Reporter:
Matt Konda
Votes:
0
Watchers:
1
Stop watching this issue
Description
Narrative: As a regular user, I need to be able to view my profile data to ensure that it contains the correct information in order to manage my status as an active user in the system.
Acceptance Criteria:
User should be able to follow a link from the top right to their profile.
User should see all relevant data (email, address, name, birthday)
Dates
Created:
13/Sep/15 5:16 PM


Example Issue for Testing Checklist
Edit
Comment
Assign
More
Open
In Progress
Resolved
Reopened
Closed
Details
Type:
New Feature
Priority:
P3
Affects Version/s:
None
Component/s:
None
Labels:
None
Status:
OPEN
(View Workflow)
Resolution:
Unresolved
Fix Version/s:
None
People
Assignee:
Unassigned
Assign to me
Reporter:
Matt Konda
Votes:
0
Watchers:
1
Stop watching this issue
Description
Narrative: As a regular user, I need to be able to view my profile data to ensure that it contains the correct information in order to manage my status as an active user in the system.
Acceptance Criteria:
User should be able to follow a link from the top right to their profile.
User should see all relevant data (email, address, name, birthday)
Security Checklist:
The page should not have XSS. It should encode any user data.
The page should not be injectable.
The page should implement authorization.
Any sensitive data should be stored / displayed in accordance with the data classification document.
Dates
Created:
13/Sep/15 5:16 PM
Updated:
13/Sep/15 5:18 PM
Drag and Drop



Example Issue for Testing Checklist

Edit | Comment | Assign | More | Open | In Progress | Resolved | Reopened | Closed | Exp

Details

Type:	New Feature	Status:	OPEN (View Workflow)
Priority:	P3	Resolution:	Unresolved
Affects Version/s:	None	Fix Version/s:	None
Component/s:	None		
Labels:	None		

Description

Narrative: As a regular user, I need to be able to view my profile data to ensure that it contains the correct information in order to manage my status as an active user in the system.

Acceptance Criteria:

- User should be able to follow a link from the top right to their profile.
- User should see all relevant data (email, address, name, birthday)
- User should only be able to see their own record.
- Viewing of a profile should be logged for future reference.
- Birthday, name and address are PII and should be stored carefully.

Security Checklist:

- The page should not have XSS. It should encode any user data.
- The page should not be injectable.
- The page should implement authorization.
- Any sensitive data should be stored / displayed in accordance with the data classification document.

Activity

People

Assignee: Unassigned
Assign to me

Reporter: Matt Konda

Votes: 0

Watchers: 1 Stop watching this issue

Dates

Created: 13/Sep/15 5:16 PM

Updated: 13/Sep/15 5:20 PM

Drag and Drop

Drop files here to attach the
or
Select files

Bug Tracking

Testing


Welcome to A Clean Well-Lighted Place for Books
415-441-6670 www.bookstore.com FAX 415-567-6885
[Home | Events | Features & Recommendations | Shopping Cart]
A CLEAN WELL-LIGHTED PLACE for BOOKS
Welcome to A Clean Well-Lighted Place for Books
Your Shopping Cart
Qty
Description
Price
Remove
Home
Events
Book Search
Autographed Books
Remainders 50% off!!
Remainders 60% off!!
Booksense 76
Linux Security for Large-Scale Enterprise Networks
Becker, Jamieson
-1
1555582923 Paperback
$-59.99
Remove
Special Order
Total: $ -59.99
Save Qty Changes
Check Out
Done
Internet
Insecure software
Secure communications

Operationalize

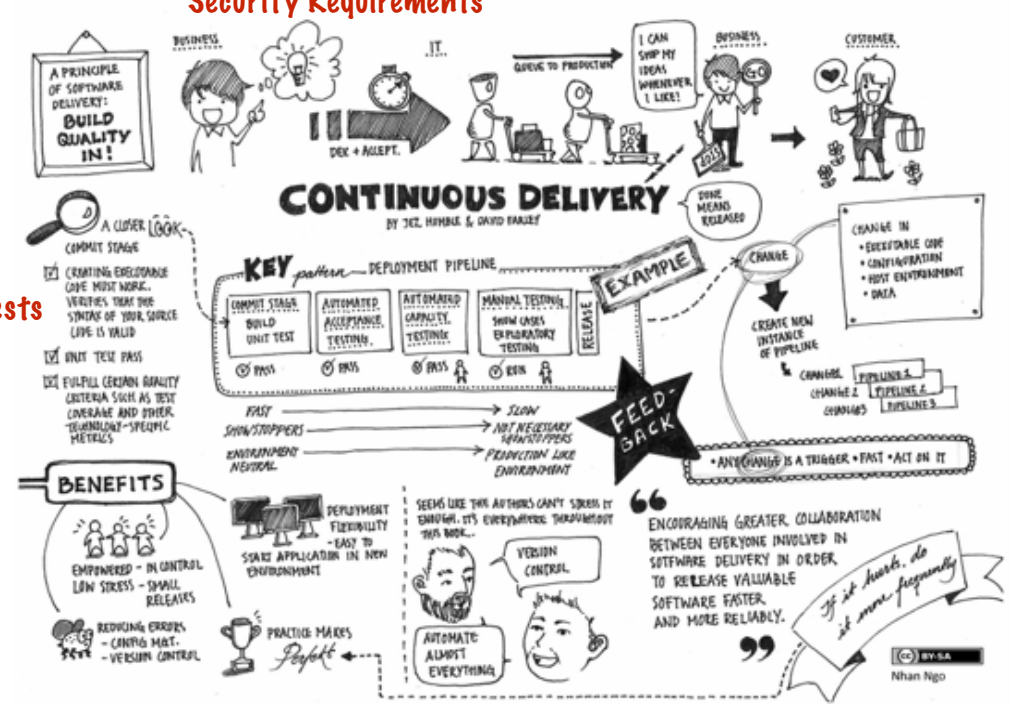
Understand lifecycle

Think incremental

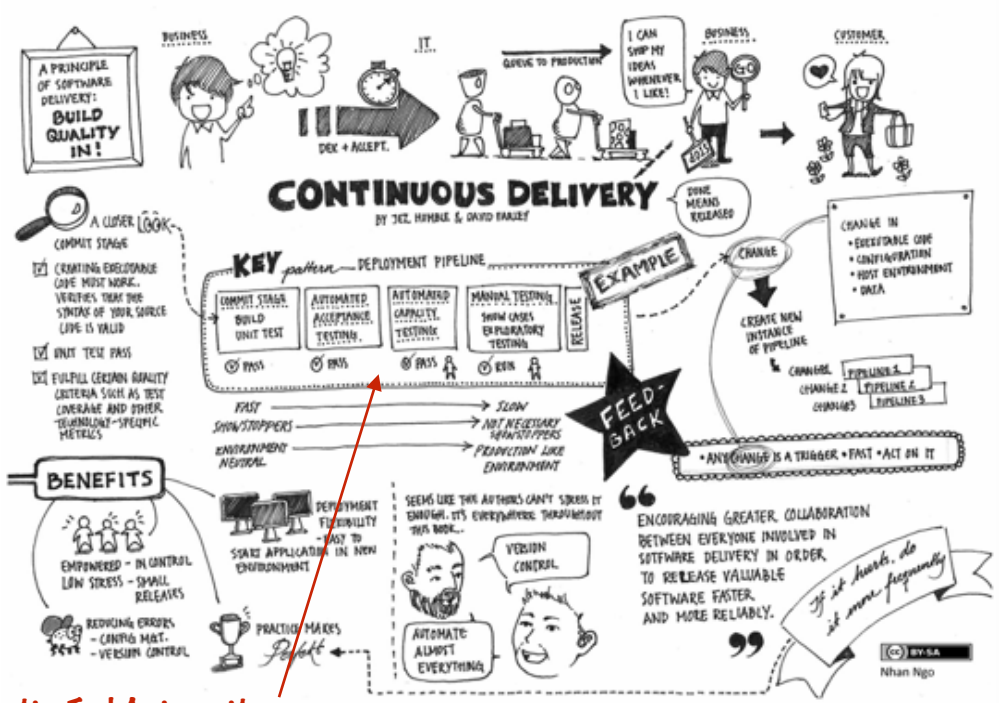
Security Requirements

Security Unit Tests

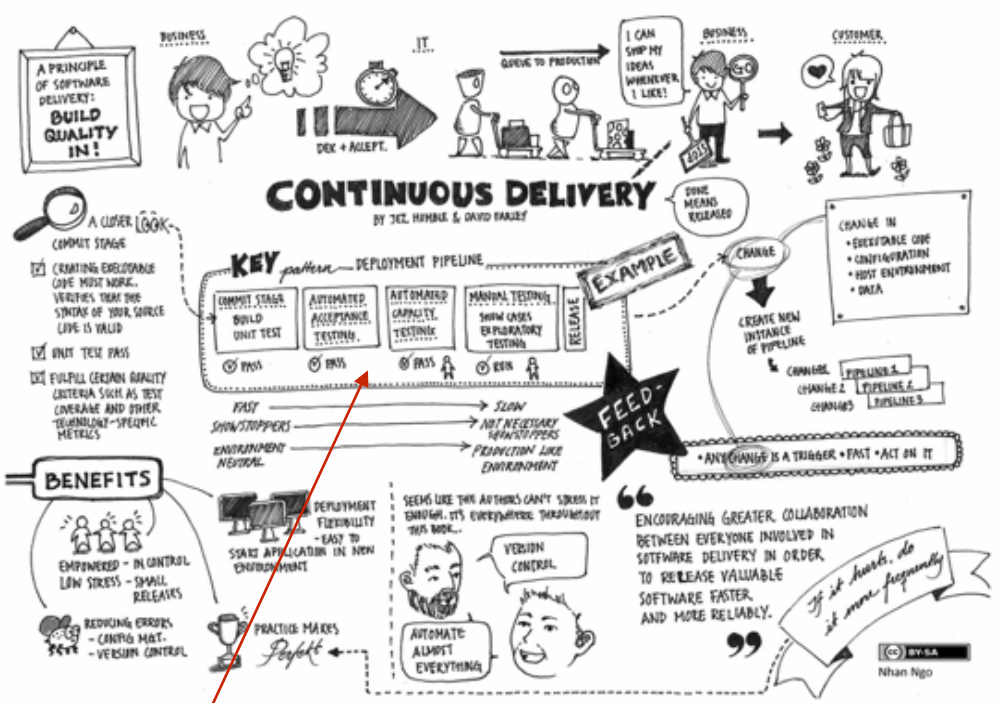
Code Review



Automate security tools



Security Tool Automation:
Code analysis
Security unit tests
Dynamic scanning
etc.



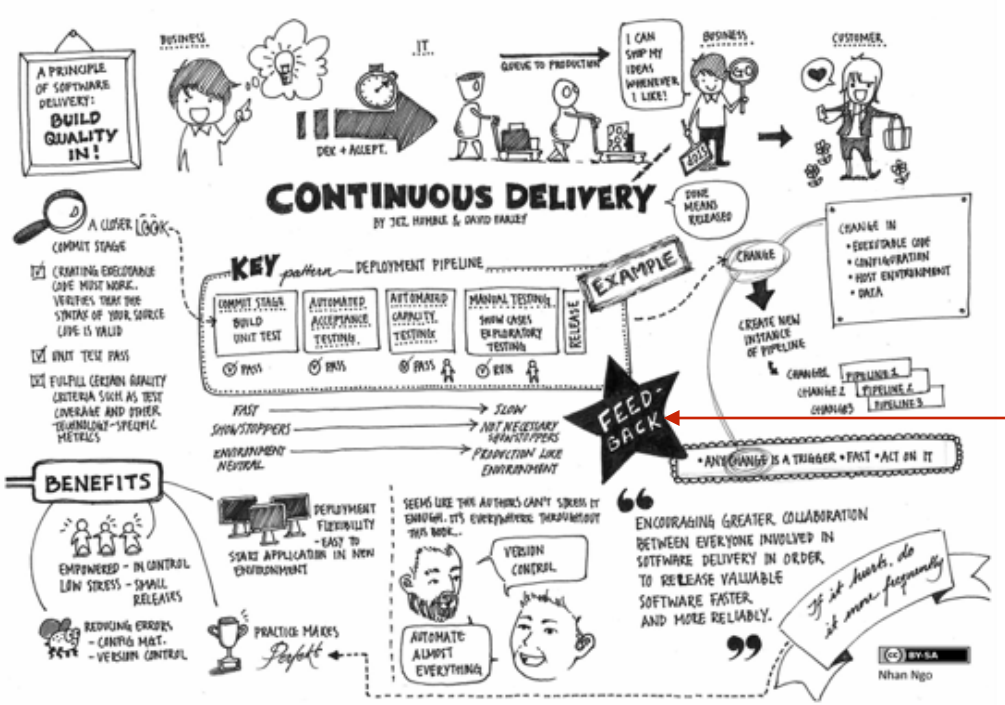
Security Tests Run

Exploratory Testing Includes Security

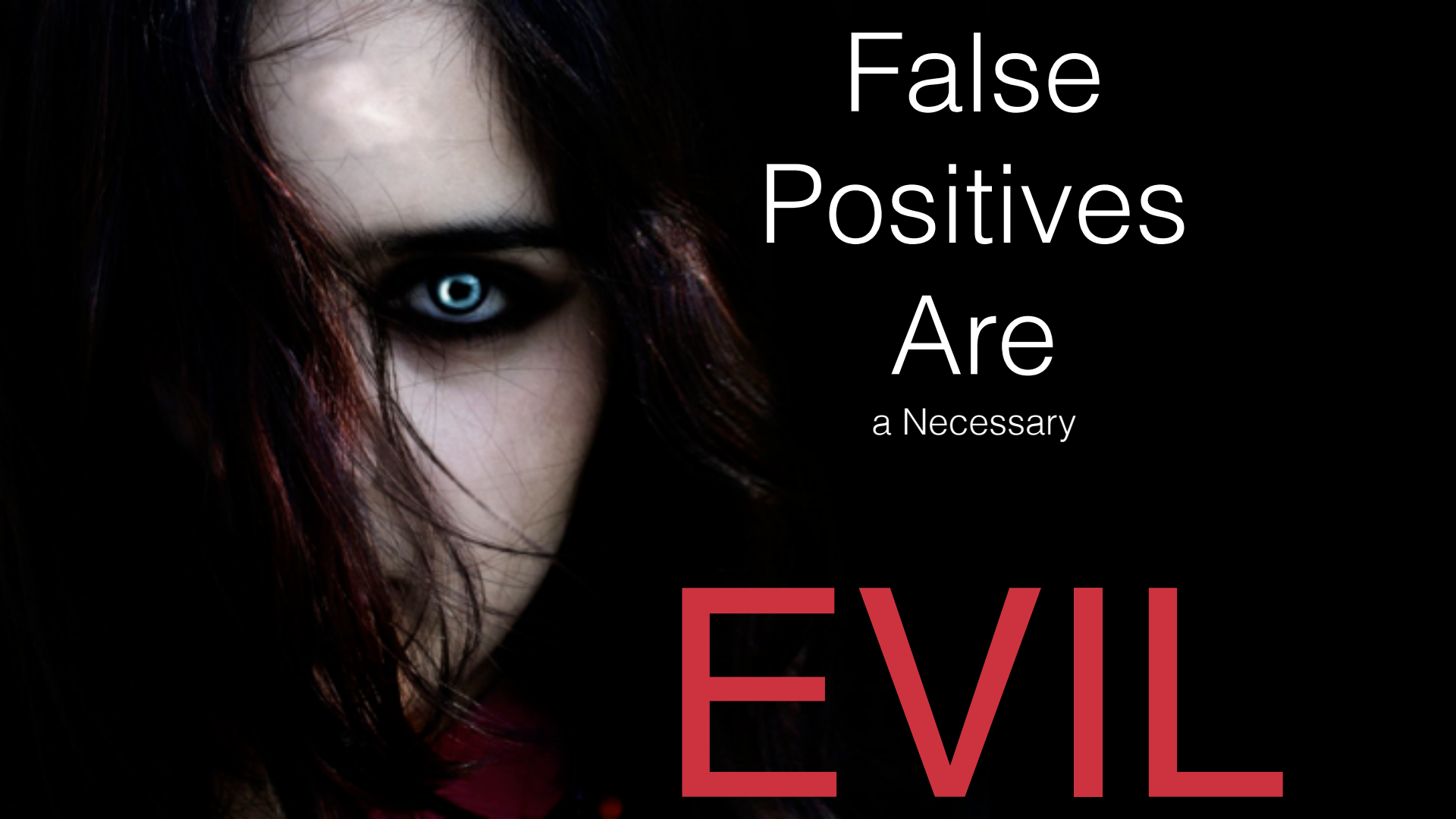
A detailed example:

- Let's say a feature is being developed
- Then devs and testers are checking a new feature
- Let them browse through an attack proxy (like Burp or ZAP) in passive mode
- At night or when the system is quiet, use the browsing pattern as seeds for overnight attacks

Continuous feedback

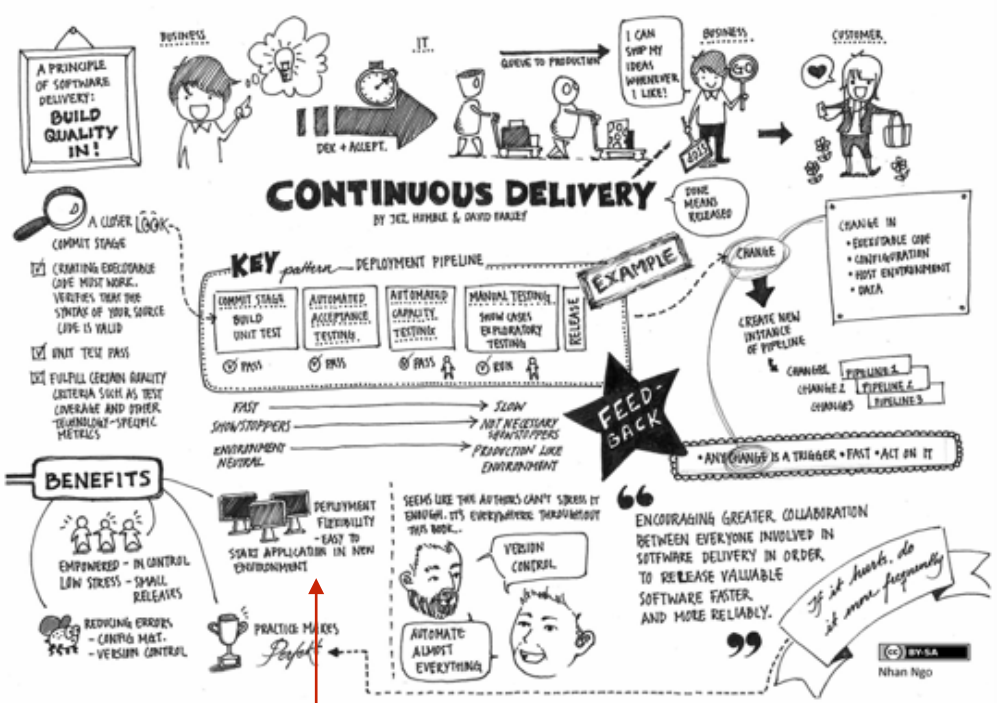

A PRINCIPLE OF SOFTWARE DELIVERY:
BUILD QUALITY IN!
BUSINESS
IT
DEV + ACCEPT.
GOING TO PRODUCTION
I CAN SHIP MY IDEAS WHENEVER I LIKE!
BUSINESS
CUSTOMER
CONTINUOUS DELIVERY
BY JEZ. HUMBLE & DAVID FARLEY
DONE MEANS RELEASED
A CLOSER LOOK
COMMIT STAGE
CREATING EXECUTABLE CODE MUST WORK.. VERIFIES THAT THE SYNTAX OF YOUR SOURCE CODE IS VALID
UNIT TEST PASS
FULFILL CERTAIN QUALITY CRITERIA SUCH AS TEST COVERAGE AND OTHER TECHNOLOGY-SPECIFIC METRICS
KEY pattern — DEPLOYMENT PIPELINE
COMMIT STAGE
BUILD
UNIT TEST
AUTOMATED ACCEPTANCE TESTING
AUTOMATED CAPACITY TESTING
MANUAL TESTING
SHOW CASES
EXPLORATORY TESTING
RELEASE
PASS
PASS
PASS
ERR
EXAMPLE
CHANGE
CHANGE IN
• EXECUTABLE CODE
• CONFIGURATION
• HOST ENVIRONMENT
• DATA
CREATE NEW INSTANCE OF PIPELINE
CHANGE1
CHANGE 2
CHANGE3
PIPELINE 1
PIPELINE 2
PIPELINE 3
FAST
SHOW STOPPERS
ENVIRONMENT NEUTRAL
SLOW
NOT NECESSARY SHOWSTOPPERS
PRODUCTION LIKE ENVIRONMENT
FEED-BACK
• ANY CHANGE IS A TRIGGER • FAST • ACT ON IT
BENEFITS
EMPOWERED - IN CONTROL
LOW STRESS - SMALL RELEASES
REDUCING ERRORS
- CONFIG MGT.
- VERSION CONTROL
DEPLOYMENT FLEXIBILITY
- EASY TO START APPLICATION IN NEW ENVIRONMENT
PRACTICE MAKES Perfect
SEEMS LIKE THIS AUTHORS CAN'T STRESS IT ENOUGH. IT'S EVERYWHERE THROUGHOUT THIS BOOK..
VERSION CONTROL
AUTOMATE ALMOST EVERYTHING
ENCOURAGING GREATER COLLABORATION BETWEEN EVERYONE INVOLVED IN SOFTWARE DELIVERY IN ORDER TO RELEASE VALUABLE SOFTWARE FASTER AND MORE RELIABLY.
If it hurts, do it more frequently
BY-SA
Nhan Ngo

Feedback!


False
Positives
Are
a Necessary
EVIL

Optimize for relevance

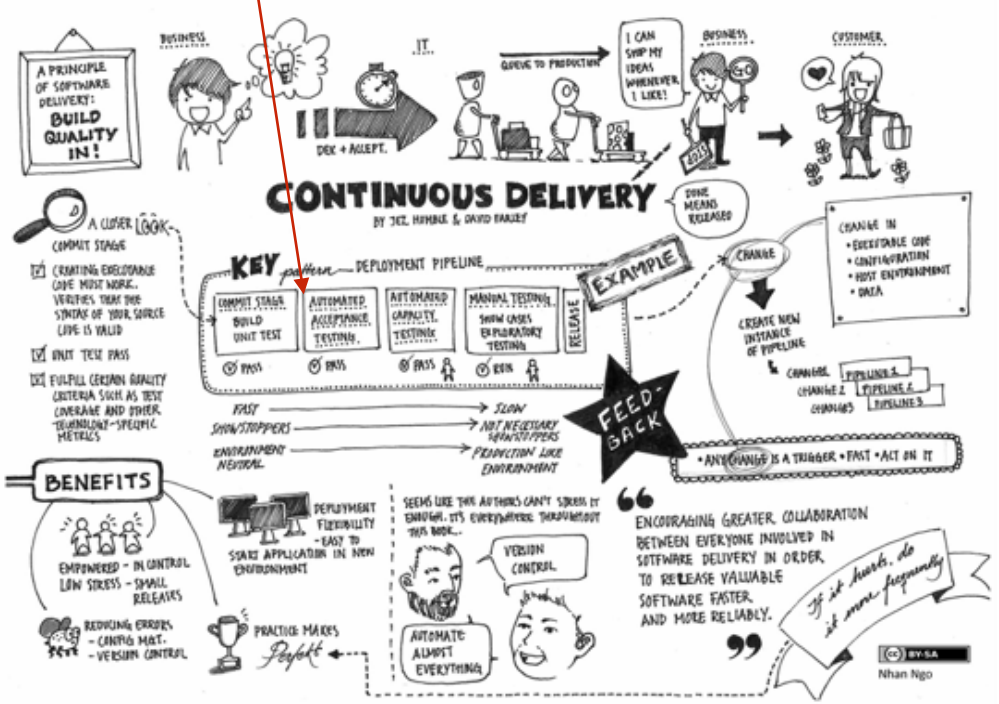
Provisioning tools



Since its easy to provision we can do security testing safely in a new env.

Audit tools

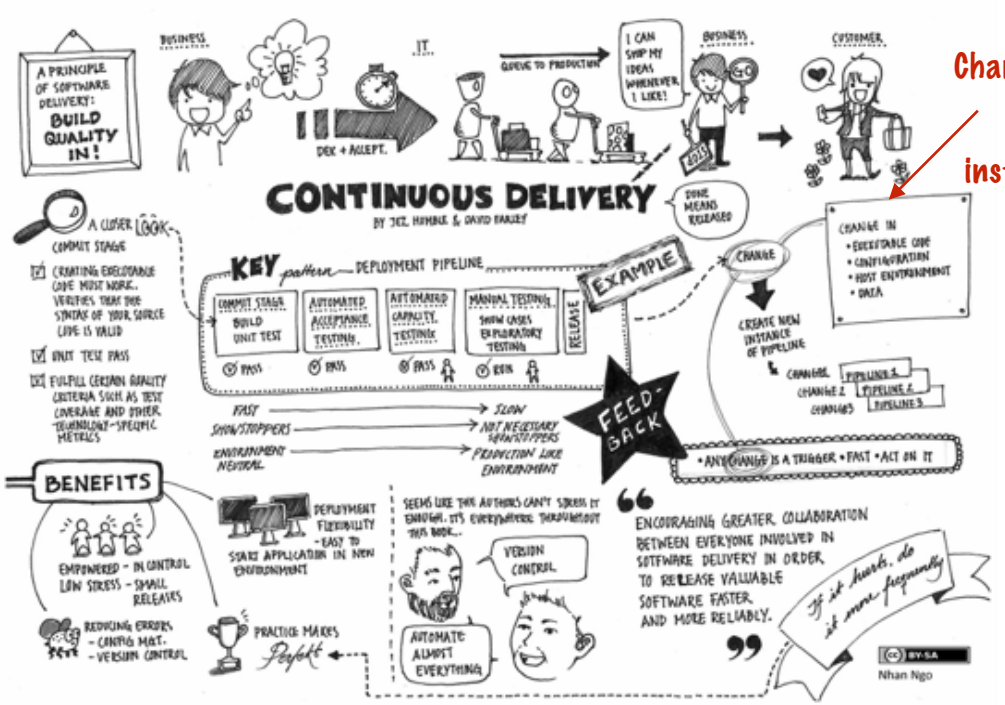
Deployment checks includes security audit checks.



Self documenting for regulatory and compliance!

Chaos tools

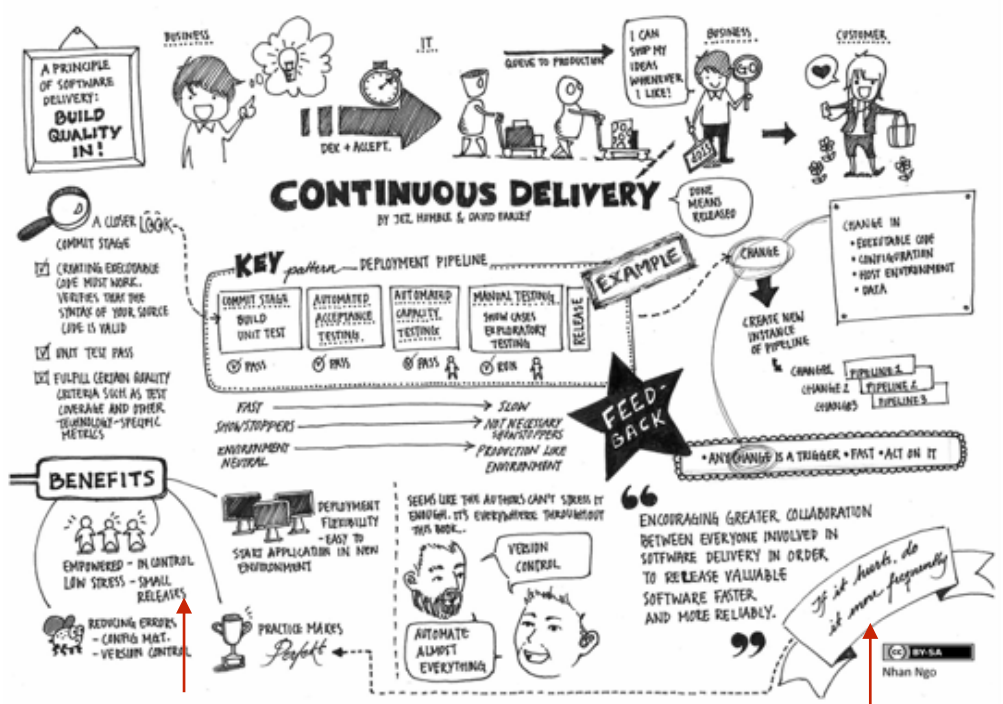
Change is good



Change is happening.
It can be an
opportunity
instead of a hassle.

Complexity is an enemy

Decomposition to micro-services reduces dependencies and complexity.

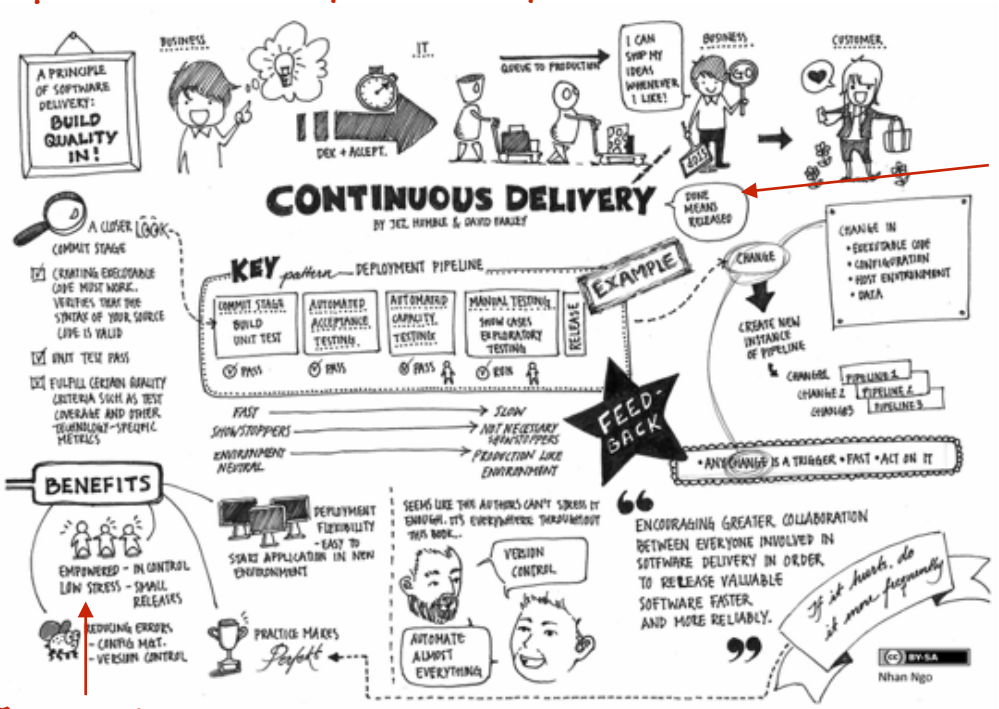


Small releases reduce complexity.

Right now, security hurts.

Shared responsibility

Another principle of software delivery: build security in!



Done means secure!

Empowered to do security right!

Measure results

Event based model … (Reactive)

Commit

- Security Unit Tests
- Static Code Analysis (Pipeline)
- Security Requirements
- Check Dependencies
- Code Review
- Checklists

Deploy

- Scripted Provisioning / Built in Change Control
- Provisioning Auditing (Chef Audit, hardening.io)
- Gauntlt

Periodic

- Full app analysis (static, manual pen test)
- Secure Development Training
- Baseline Security Requirements Review
- ASVS Review
- Data Science on Results

Security Incident

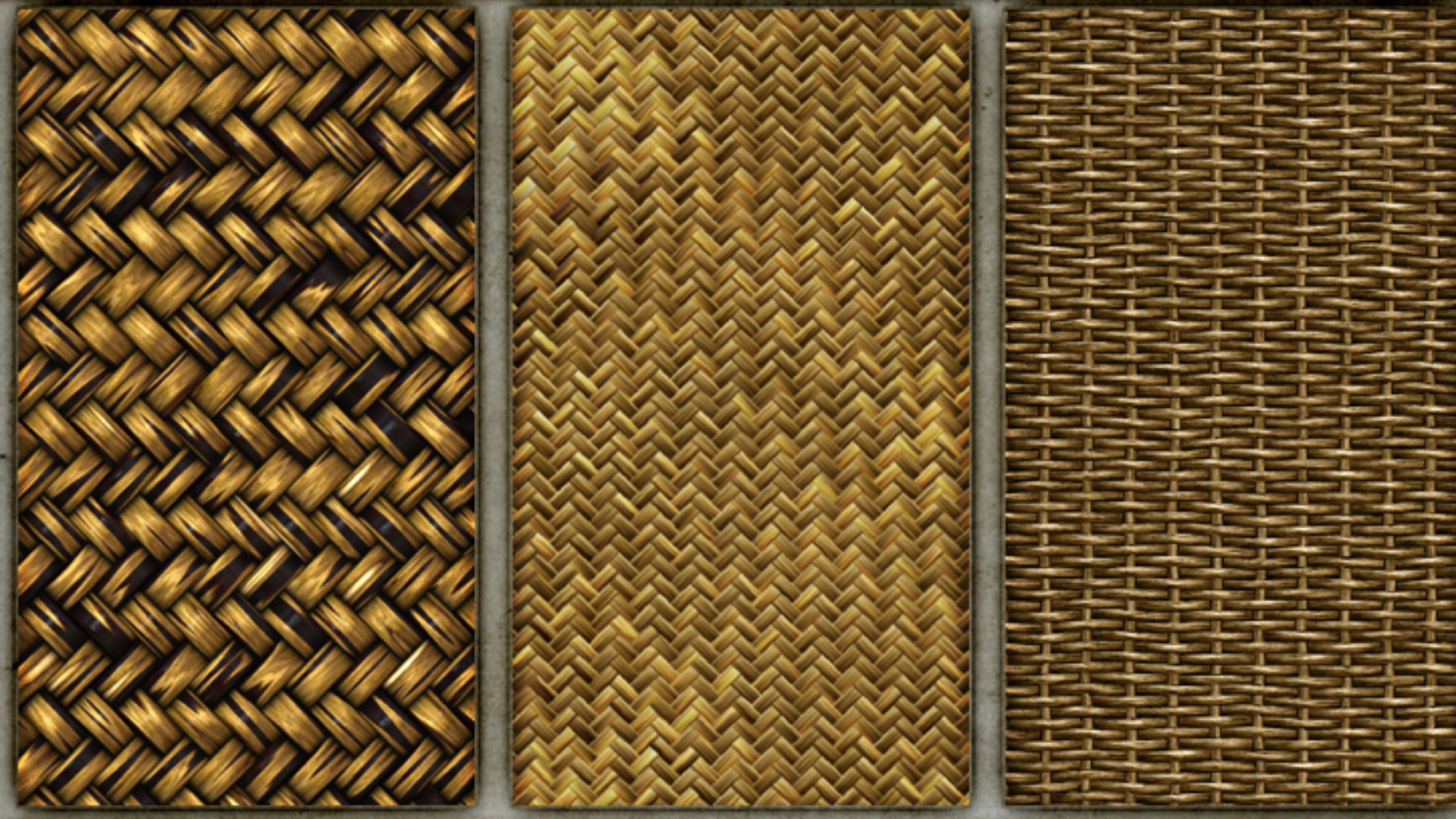
Required metasploit struts demo…

```
/bin/dependency-check.sh          -
a struts2-showcase                -
out /tmp/                          -
s /tomcat-root/struts2-showcase/
```

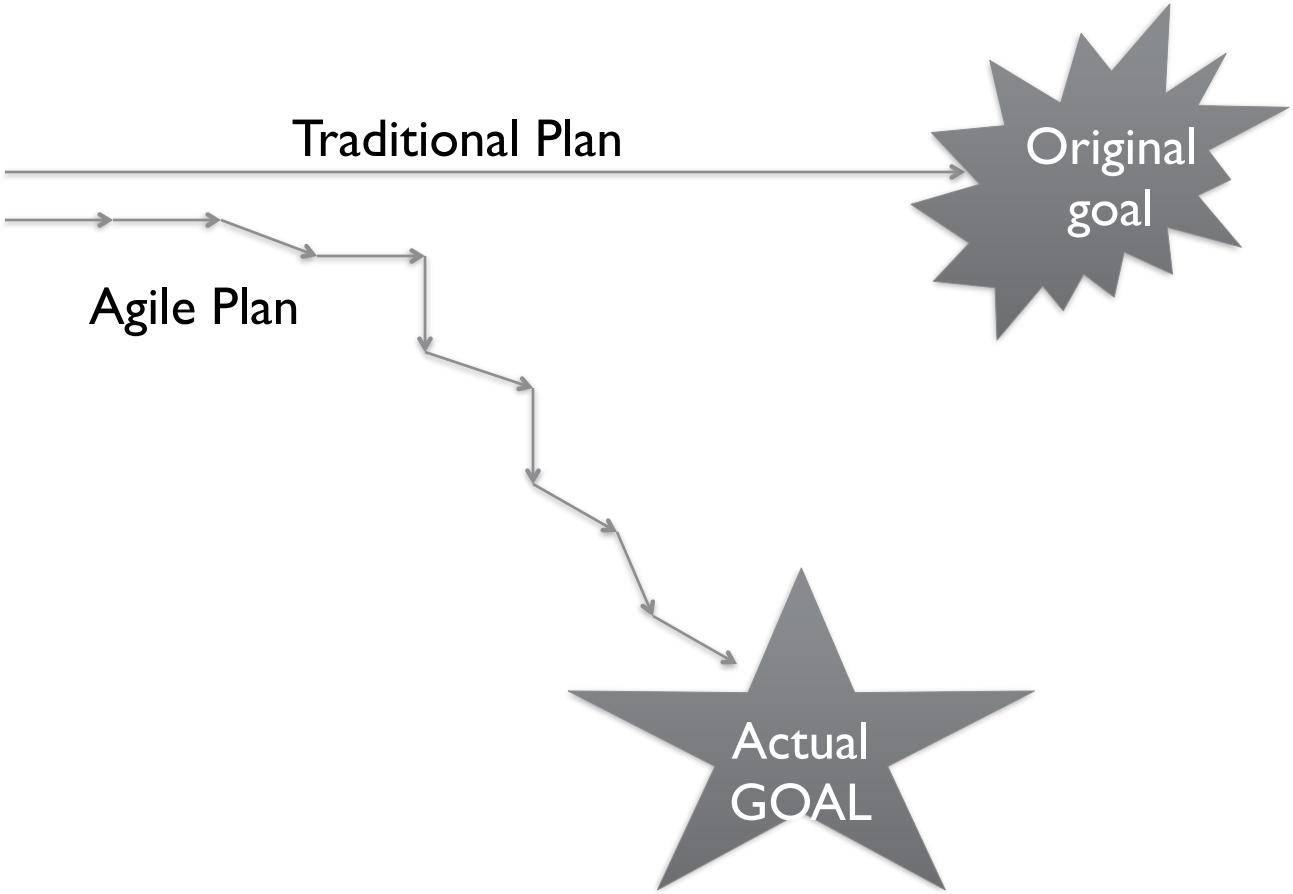
```
 <vulnerability>
   <name>CVE-2013-2251</name>
   <cvssScore>9.3</cvssScore>
   <severity>High</severity>
   <cwe>CWE-20 Improper Input Validation</cwe>
   <description>Apache Struts 2.0.0 through 2.3.15 allows remote
attackers to execute arbitrary OGNL expressions via a parameter
with a crafted (1) action:, (2) redirect:, or (3) redirectAction:
prefix.</description>
   <references>
    <reference>
    <source>BID</source>
      <url>http://www.securityfocus.com/bid/64758</url>
      <name>64758</name>
   </reference>
   …
```

Takeaway: lots of your issues might be in your dependencies!

So what *is* Rugged all About?



Traditional Plan


Traditional Plan
Original goal
Agile Plan
Actual GOAL


empathy


EXIT 23
ACCOUNTABILITY
STRAIGHT AHEAD


culture

CHICAGO
INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE 2016

goto; conference

Please

Remember to rate this session

Thank you!