# Secure by Design

## Building secure and useable systems in a connected world


Laura Bell

FOUNDER & LEAD CONSULTANT
SAFESTACK

@lady_nerd
laura@safestack.io

the world is a terrible place

the internet is a festering pool of toxic waste

somebody probably wants to do bad things to your computer

the security situation is beyond hope

we can build amazing things

doing this securely is hard

# In today's talk….

Security is an essential skill for all developers and testers

Five simple tools and techniques that can bring security to any dev/test environment

Bringing security to your world can be challenging but you can avoid common pitfalls.

# Security for everyone

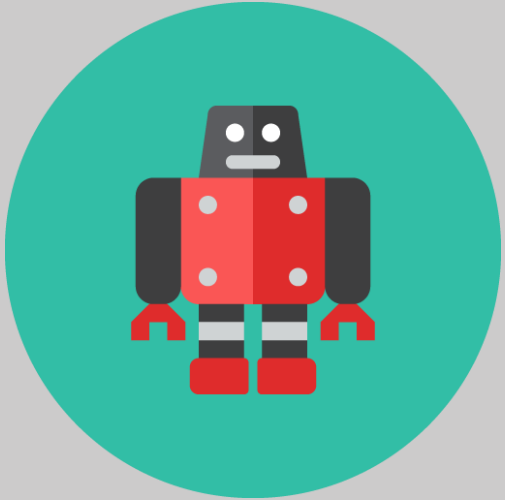
# Security needs a hero

security is a team sport

SafeStack.io

# Security is a tester superpower

security starts with education



# Tools and Techniques

1. Requirements capture
2. Language
3. Design level security
4. Automation
5. Consistency

# Requirements Capture

security starts with requirements

Functional/Story requirements:

User enters username on forgotten password page

User receives link to reset password page

Password is reset

User can login to system

# Forgotten Password Example

Security requirements:

Password reset link expires after 24 hours

Password reset link is unique to password reset request

Password reset link is complex and pseudo random

Password reset link can only be used once

Error messages on password reset form do not allow username or email enumeration

# Forgotten Password Example

# Language

Language creates barriers and divides

Critical

High

Medium

Low

Informational

False Positive

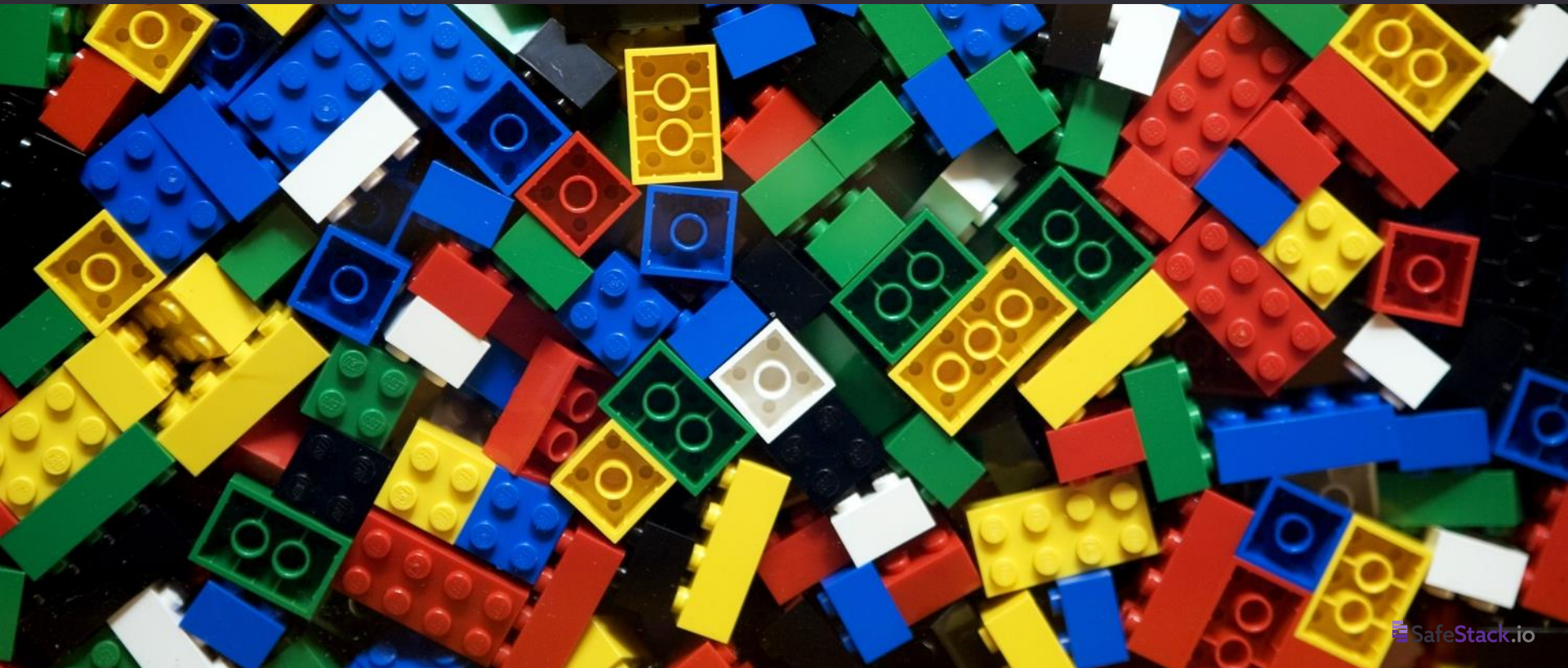
# Get the language right

Security vulnerabilities hide behind acronyms, jargon and assumptions.

# Design level security

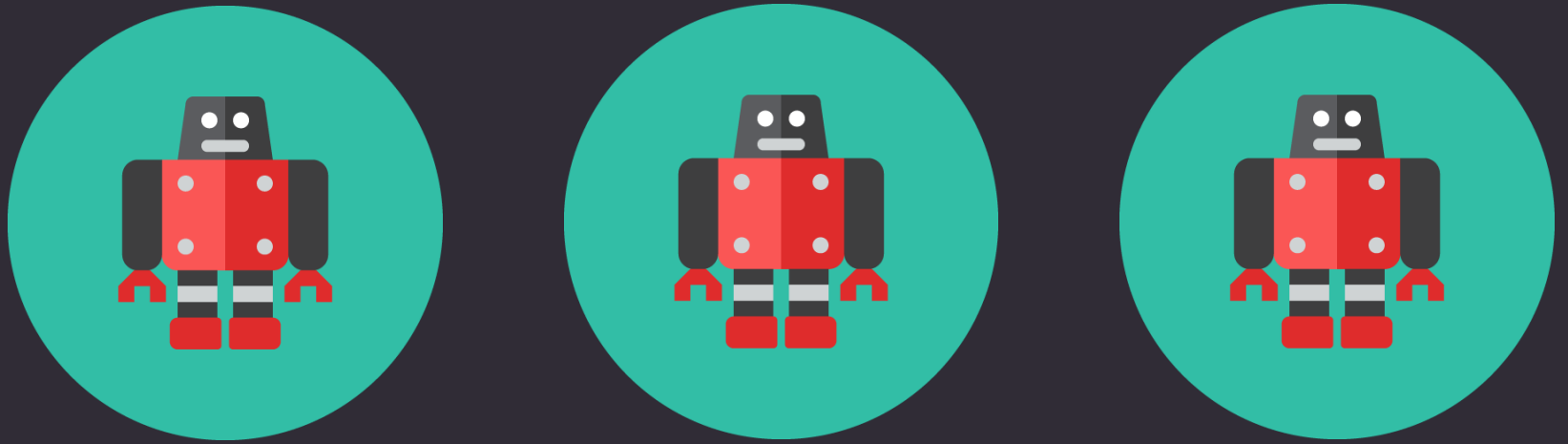
Vulnerabilities cluster between components

# Our environments are complex



Bad things happen when low risk security issues cluster together.....

# Automation



Human error threatens testing effectiveness

Test data creation and scrubbing
Test case definition
Load testing
Regression testing
Integration testing
Security testing
Test environment deployment

Automated security testing gives a continuous assessment of risk.

# Consistency

Don’t leave security to chance

All testers

All stories

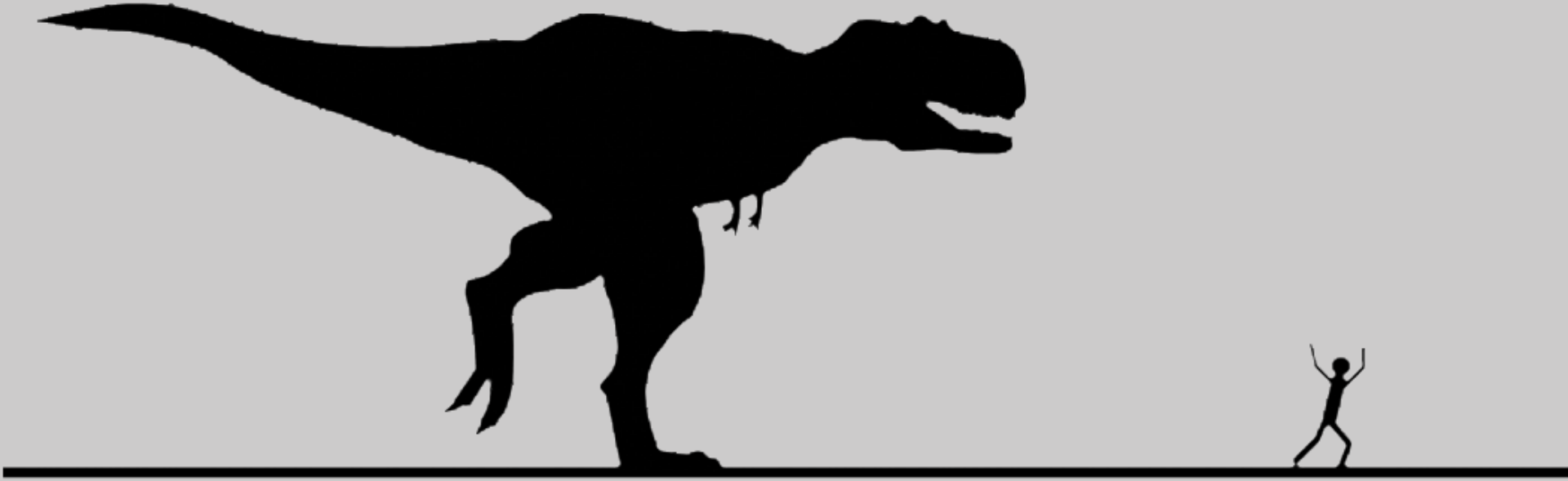
All the time

1. Requirements capture
2. Language
3. Design level security
4. Automation
5. Consistency

# Common Challenges

1. Avoid security theatre
2. Stop ignoring legacy code
3. Maintain momentum
4. Face your fear

# Avoid security theatre



# Dealing with a legacy

# Aim for continuous steady improvement

# It’s OK to be afraid

1. Avoid security theatre
2. Stop ignoring legacy code
3. Maintain momentum
4. Face your fear

# TL;DR

Security is an essential skill for all developers and testers

Five simple tools and techniques that can bring security to any dev/test environment

Bringing security to your world can be challenging but you can avoid common pitfalls.

# Questions?


Laura Bell

FOUNDER & LEAD CONSULTANT
SAFESTACK

@lady_nerd
laura@safestack.io