Developing secure internet applications in a dangerously connected world

Frank Zinghini

frank.zinghini@avi.com

The Threat

Why all the fuss about cybersecurity?

- *2,100 confirmed data breaches in 2014*

  Source: Verizon Data Breach Report (VDBR)

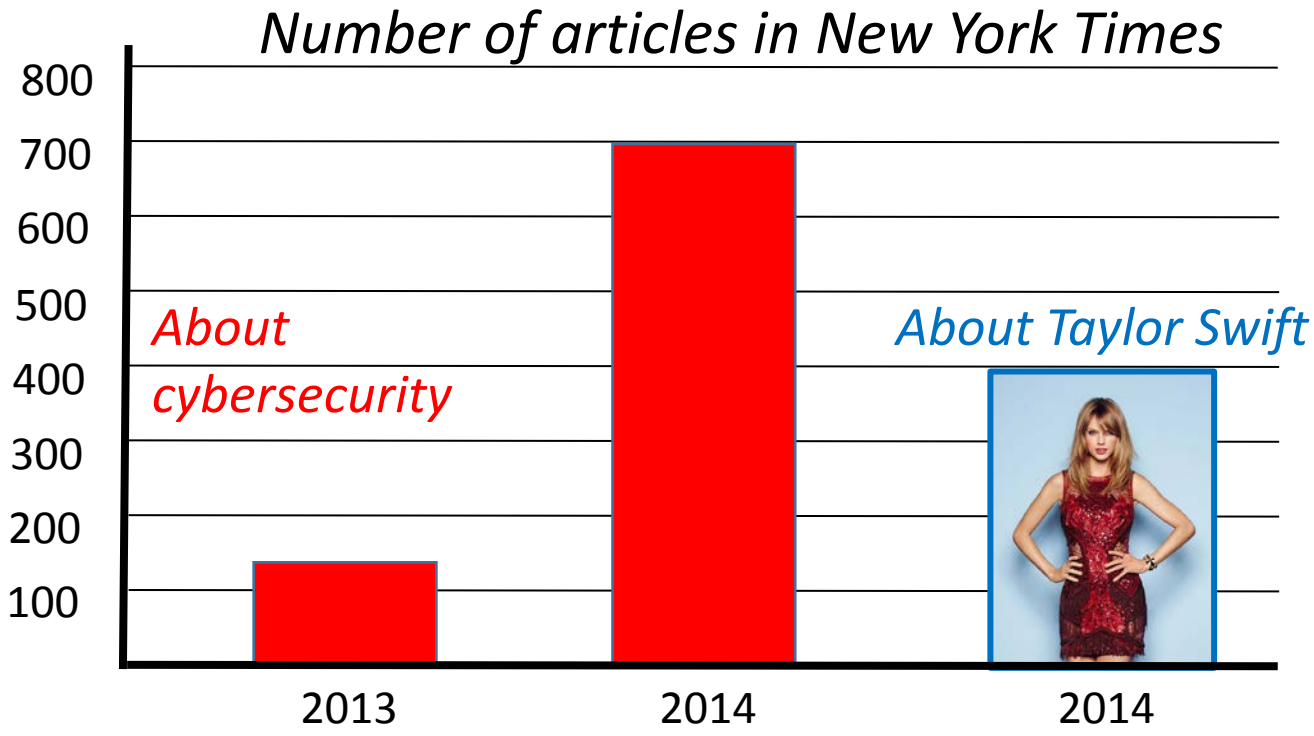
- Average cost per company: $3.5 million

  Ponemon 2014 Cost of Data Breach Study: Global Analysis, 2014

- Average cost *per data record* breached: *$201*

- How do you measure impact on reputation?

Cybersecurity is trending



What is encompassed by cybersecurity?

- The **attackers**, and their **modus operandi**
- **Entities**, part of computing infrastructure being targeted
  - Networks – Wired, wireless
  - Devices – Workstations, laptops, tablets & phones
  - Applications – Apps you build, buy, and download; Your web site
  - Internet of Things
  - Your own people
- **Defensive** mechanisms
  - Technology/people to *detect*, *fend off* and *respond to* attacks
  - **Forensics investigations** to determine *first instance* and *sequence of events* leading up to and following from incident.
- **Preventive** mechanisms to *remove attacker opportunities*

Attackers: Who are they and what do they want?

Fun Script Kiddies strutting their stuff
Fame Hackers with poor moral compasses
Fortune Plain old criminals seeking wealth; just another channel for their crimes
Force Political

- Nation-state cyber soldiers – "APT"
  - Exerting political influence, stealing trade secrets
  - Known bad actors: China, N Korea, Russia, Estonia, Iran
- Terrorists/Saboteurs
  - SCADA (e.g. power plant) attacks, changes in financial market, EMT car hacks
  - 2010 FBI discovered Russian malware inserted into NASDAQ
- Hacktivists
  - Pushing a political agenda via cyber means

http://www.bloomberg.com/bw/articles/2014-07-17/how-russian-hackers-stole-the-nasdaq

Cybercrime: Plain old criminals

> This year, organized crime became the most frequently seen threat actor for Web App Attacks.
>
> Verizon Data Breach Report 2015

Web Applications are used to perpetrate **31% of breaches into Financial Services**

What are they after?

- Databases
  - Intellectual property
  - Customer data: ebay 145m records stolen
- Communications
  - Email archives SONY PICTURES
  - Blackmail, trade secrets
- Funds transfers
  - About *half of broker-dealers and investment advisers* receive fraudulent emails to transfer funds*
- Leverage
  - Ransomware – Cryptolocker

**2/3 reported these to the Financial Crimes Enforcement Network (FinCen)*

Who are the victims?

Top 3 targeted industries:

1. Public sector
2. Information (publishers, software, media)
3. Financial services

Source: Verizon Data Breach Report

70% of targeted attacks have *secondary victims*

Example: Primary target is a database of financial records; Secondary target is the web site used to get to the database

Financial sector is one of the largest targets

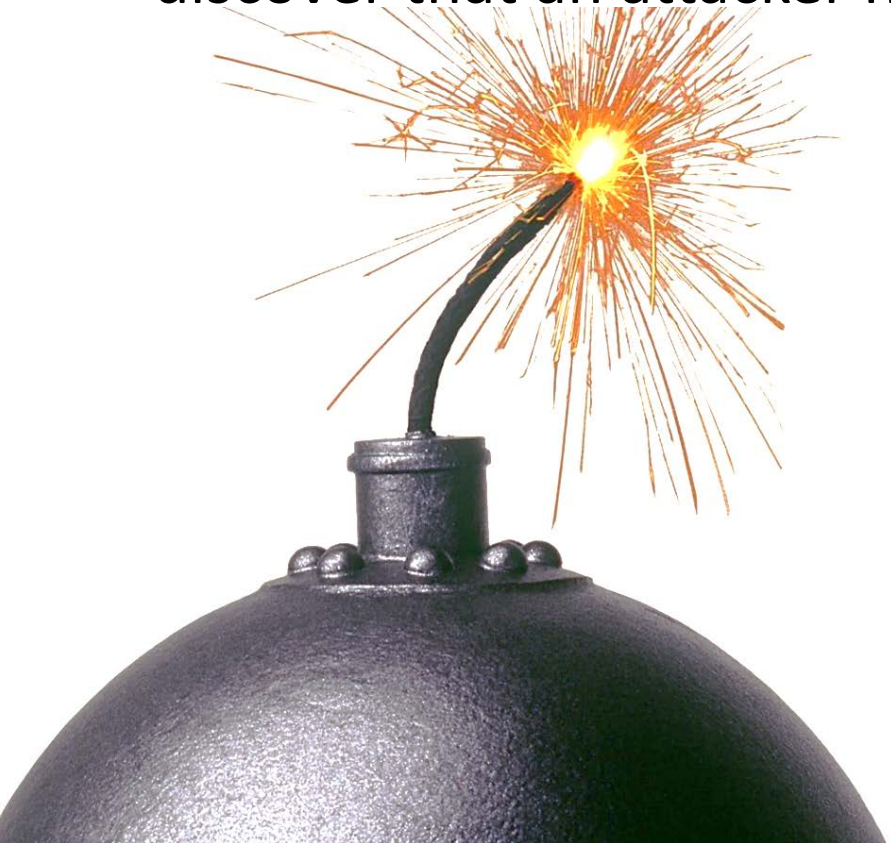
- **88%** of broker dealers and **74%** of investment advisers experienced cyber attacks *directly* or *through vendors*

  Source: Office of Compliance Inspections and Examinations (OCIE)

- **Web Applications** are used to perpetrate **31% of breaches** into **Financial Services**

How much time to detect and respond to an incident?

- In 60% of reported cases, **attackers** were able to compromise an organization within **minutes**
- On average, it takes **defenders** over **200 days** before they discover that an attacker has been in their enterprise



Cyberspace is different:

- Criminal can be in two places at once
- Stolen data doesn't disappear
- Many of the steps to conduct a crime are automated

How do they get in?

Networks

- Wired, wireless
- Break in through external connectivity

Devices

- Workstations, laptops, tablets & phones
- Get onto a device, then get into the network

Applications – most cybersecurity incidents traceable to software vulnerabilities

- The software that you expose to the internet, including your web site
- Apps you build, apps you buy, and code you download for free

Internet of Things

- Growth in connected devices creates many more ways in

People

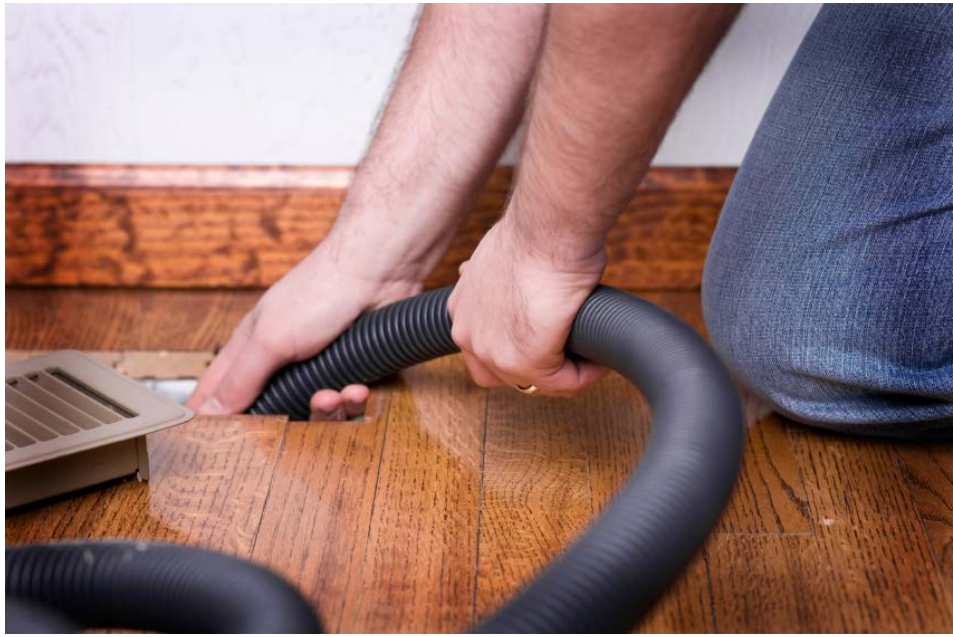
- Hardest to control: they keep leaving the doors unlocked

Indirect wired or wireless network access

Sucking in confidential corporate data through a vendor network

TARGET

Breach began with stolen credentials from **Fazio Mechanical Services**, who had access to Target's network to monitor air conditioning



Lesson: Require vendors to adhere to your cybersecurity policies

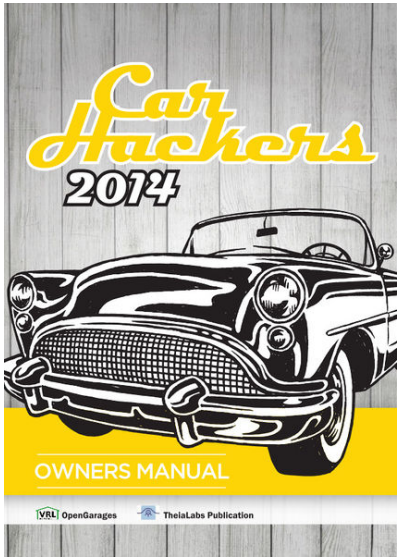
Devices

- Cell phones, tablets, other BYOD (bring your own device)
  - 96% of cell phone malware is targeted at **Android** devices
  - More than **5 billion** downloaded Android apps vulnerable to remote attacks
  - Most cellphone malware is “adnoyance”; but can be used to carry **other payloads** to help gain access to the enterprise
- Thumb drives / USB flash drives
  - Spread Stuxnet worm that attacks industrial centrifuges
  - Used in 2008 attack on US DoD; took 14 months to clean up.
- Infiltrated laptops
  - 350,000 Morgan Stanley client accounts an adviser had downloaded onto his unsecured laptop

http://www.cnbc.com/id/102462850

The Internet of Things (IoT)

- Smart Appliances
  - 2013 spam attack– Attackers introduced malicious software into networked **refrigerators** to dispense spam messages
- Automobiles – New cars are networks on wheels
  - Can hack into speedometer, acceleration, cruise control, braking, on board navigation
  - Infotainment centers are an easy access point
  - Check out www.youtube.com/watch?v=lpZulZ1rFmY
- Baby monitors sound and video have been hijacked
- "Smart toilets" – Free Android phone app can remotely trigger flushing and bidet functions on Japanese "Satis" toilets via Bluetooth



http://www.infowars.com/for-the-first-time-hackers-have-used-a-refrigerator-to-attack-businesses

http://www.theawl.com/2013/08/is-your-japanese-smart-toilet-trying-to-kill-you

What's different about Dick Cheney's pacemaker?

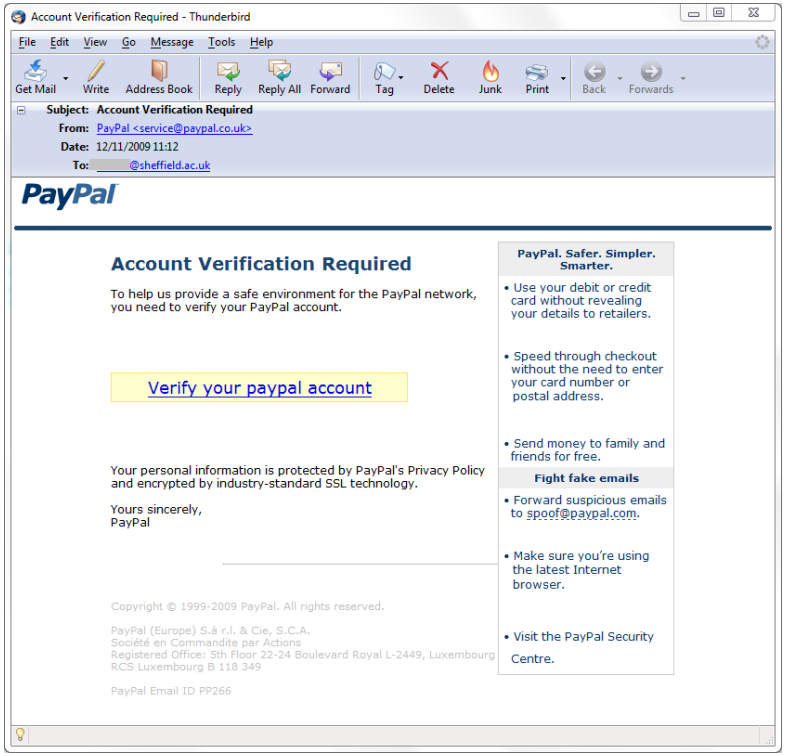
People make poor decisions

- Neiman Marcus decision to slow-roll smart chip technology in credit cards – “moral hazard”
- Network defender ignored alerts signaling intrusion into TARGET
- SONY PICTURES staffer moved confidential records to unauthorized system
- IT Tech at JPMorganChase failed to configure server with two-factor authentication, leading to data breach. Tech was from *recently acquired* company.
- Phishing expeditions are part of many attacks

People fall for phishing

Phishers trick you into handing over sensitive information by pretending to be someone they are not.

Exercise extreme caution in responding to any email asking you for personal or corporate information.



Unfortunately, phishing works…

23% of recipients open phishing messages; esp. Communications, Legal & Customer Service

11% click on attachments in those messages

66% of cyber espionage incidents use phishing

Carbanak Gang used spearfishing this year to penetrate 100 financial institutions; even accessing video feeds within "secured" office spaces

Source: Verizon Data Breach Report (VDBR)

Software flaws are at the root of *most* cyber incidents

90% of security incidents result from **exploits** against **defects** in software

Build Security In website, DHS
https://buildsecurityin.us-cert.gov/bsi/mission.html

Bug Bounties

Google pays "white hat" hackers up to $20k to find vulnerabilities in its Web browser, before attackers do

Microsoft offers as much as $150k

United Airlines pays bounties in *air miles*

What can *you* do about it?

Defending the castle

Network defense

- IDS, IPS, firewalls, antivirus

Device defense

- Password protection, policy management, system patching, two-factur authentication

Application defense

- Finding vulnerabilities in your software as you develop it
- Impose security standards on the software that you purchase
- Implement an application firewall

Shared threat intelligence

- FS-ISAC; new formats for automating shared intelligence

User defense

- User education; security culture


Application
Security

Bad, bad code…

There may be things in your code that an attacker can use to his advantage…sometimes, many things:

SQL Injection

Cross-Site Scripting (XSS)

Cross-Site Request Forgery (CSRF)

Buffer overflows

Using hard coded passwords

Weak encryption

Sensitive data exposure

So many more …

Terminology

Weakness

- A piece of code that *might* be used by an attacker in some way

Vulnerability (“*vulns*”)

- A *weakness* that can be shown to be *exploitable* by an attacker

Exploit

- What an attacker can achieve (and probably has) using that vulnerability

Triage

- The arduous task of classifying weaknesses as either vulns or false positives, and prioritizing those vulns for *remediation*

A weakness becomes a vulnerability

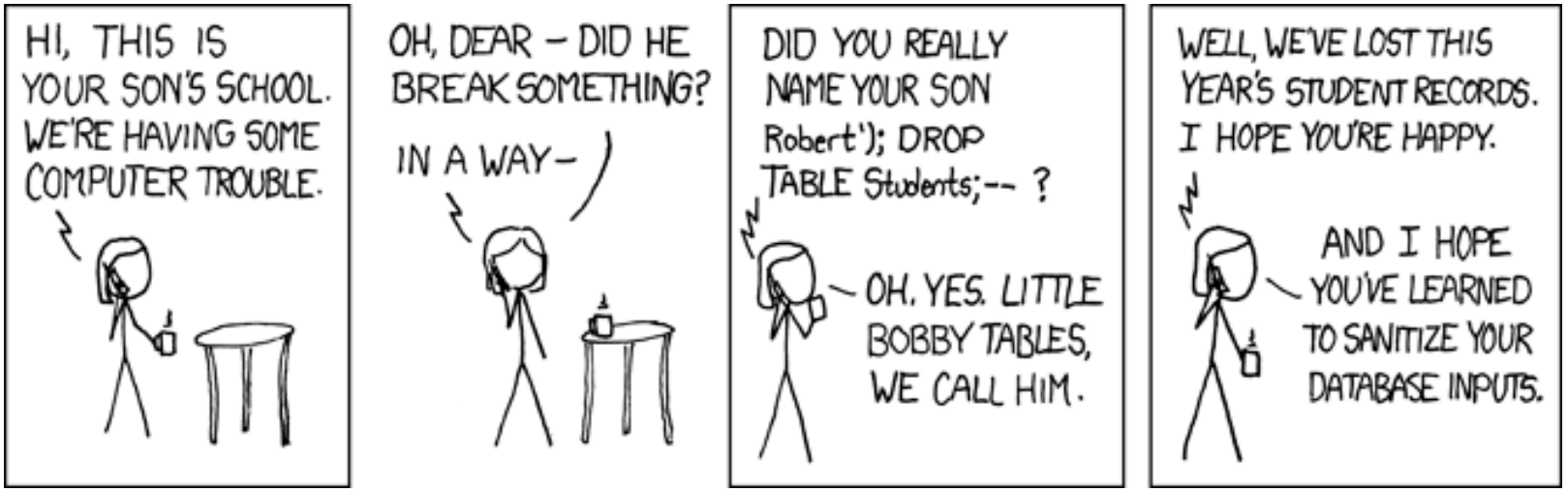
Can the weakness code be exercised by attacker action?

If so, can the attacker get any utility from exercising the weakness?

Yes? It's a vulnerability; else *false positive*

*You must understand your **threat model***

Sample exploit – SQL injecton



Blair	Hester	bhester	bhester@hyperic.com
bruce	snyder	bsnyder	bob@bob.com
Chip	Disabled	cdisabled	chip.witt@springsource.com
chris	harris	charris	chris.harris@springsource.com
chris	prendergast	chrisp	chris.prendergast@hyperic.com
Charles	Lee	clee	charles.lee@hyperic.com
colin	Sampaleanu	colin	Colin.Sampaleanu@springsource.com
Contegix	Customer	contegix	greg.walters@contegix.com
chris	prendergast	cprendergast	Chris.Prendergast@hyperic.com

NEW... DELETE Total: 117 Items Per Page: 15 1 2 3 4 5 6 7 8

Application security – approach

Review the source code

- There are known coding patterns that present potential weaknesses
  - Common Weakness Enumeration (CWE) [MITRE – www.cwevis.org]
- Manual code review reveals some
- **Static Application Security Testing** (SAST) tools find them automatically

Simulated attack

- Probe a running application looking for ways in – *black-box testing*
- Apply known attack patterns looking for response *Pen-Testing*
- **Dynamic Application Security Testing** (DAST) tools act as "robot hackers"

Types of Application Security Testing

Manual

- Code review
- Manual pen-testing
- Low false positives
- Finds things that automated tools miss
- Hard to scale

SAST

- Finds theoretical issues
- High false positives
- Results need to be triaged
- Automated, scalable

DAST

- Requires a running application
- Often used later in development cycle
- Automated, scalable

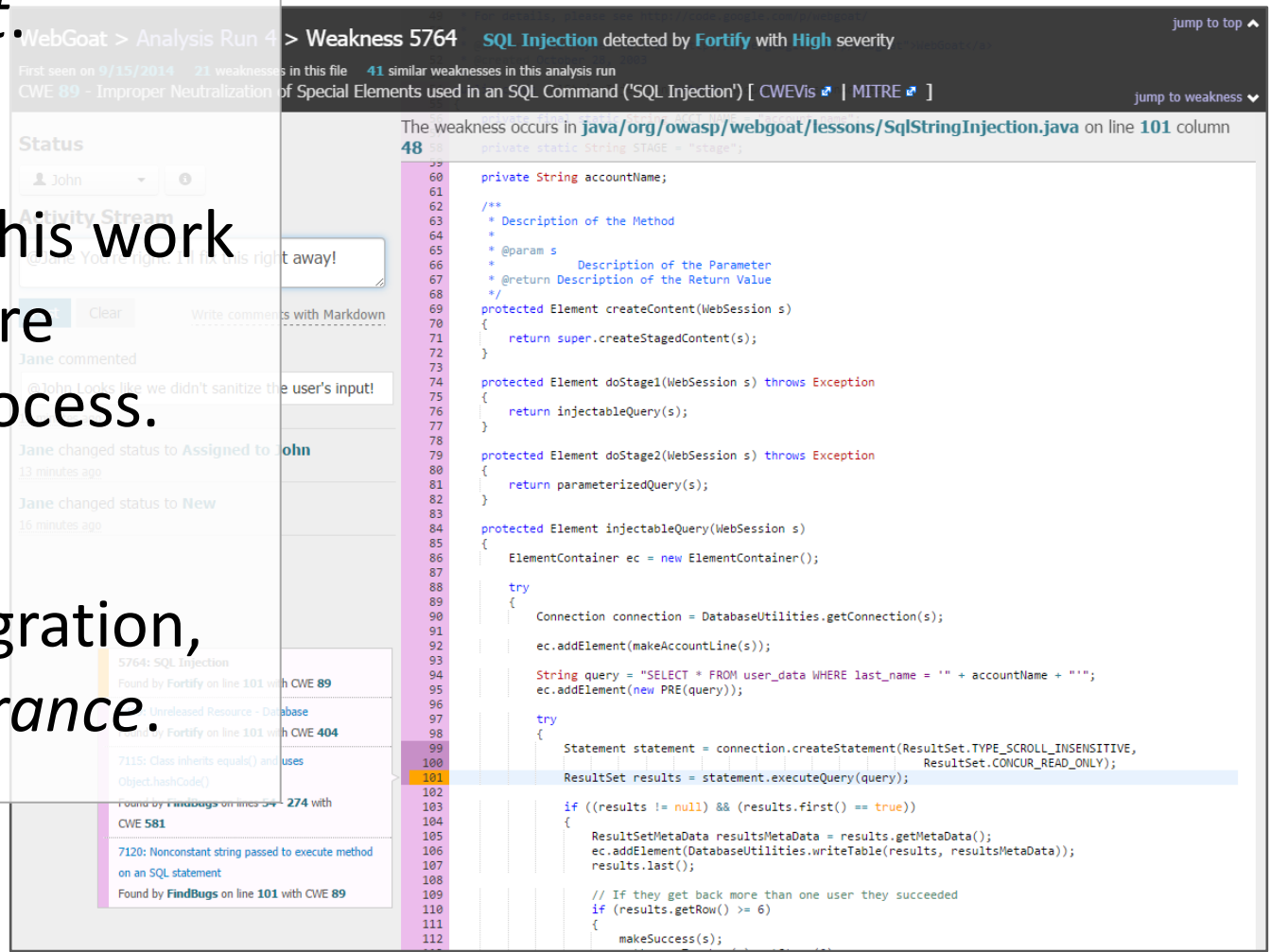
Doing the work

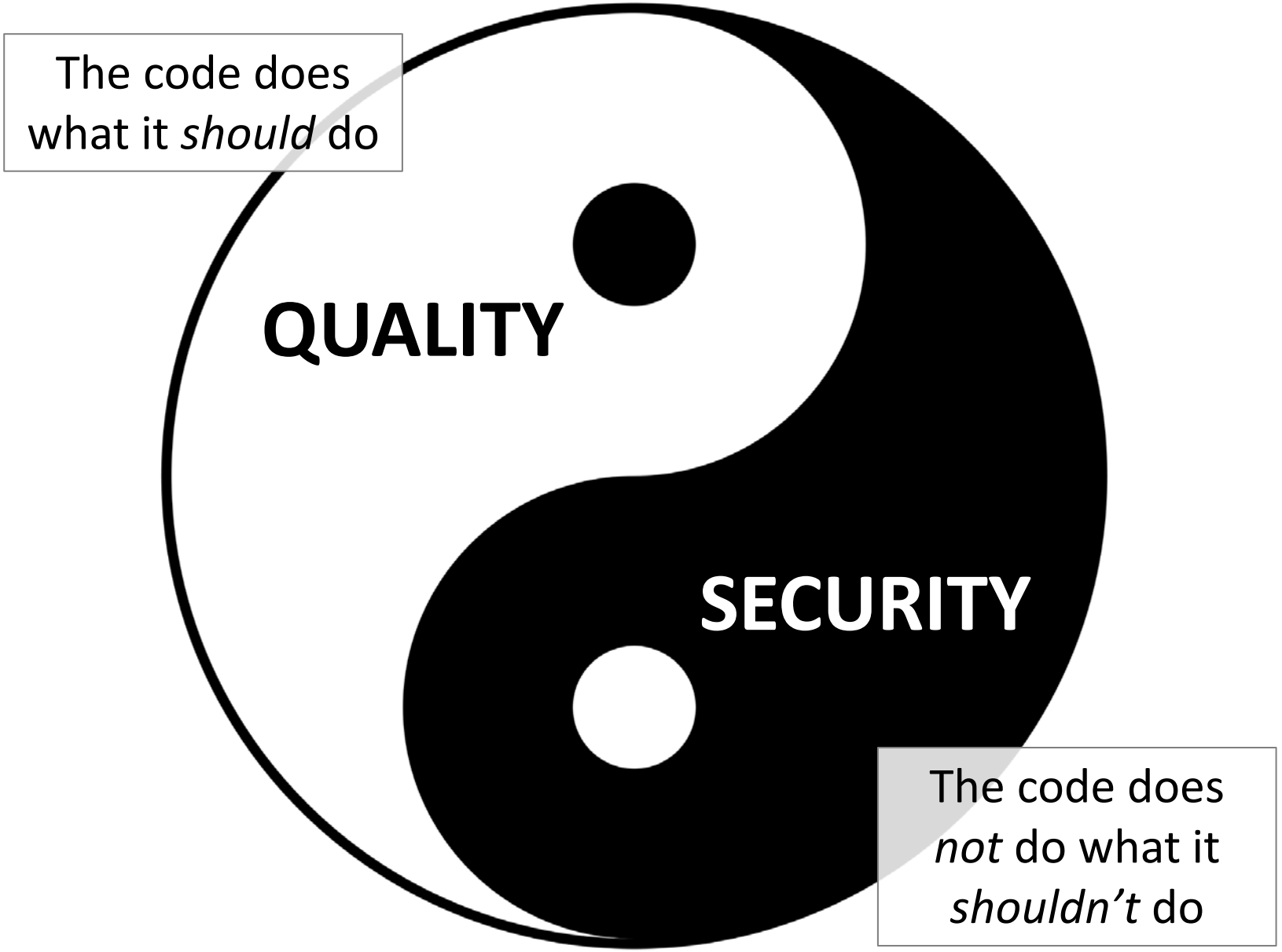
Find it, validate it, fix it.

Rinse and repeat.

You **must** build this work into your software development process.

Continuous integration, *continuous assurance*.




The code does what it *should* do
QUALITY
SECURITY
The code does *not* do what it *shouldn't* do

Quality and *Security*

Quality Issues

Confusing code

Performance issues

Concurrency issues

Memory leaks

Null pointer

Infinite loops

Redundant & dead code

… and more

Security Issues

SQL Injection

XSS

CSRF

Buffer overflows

Using hard coded passwords

Weak encryption

Sensitive data exposure

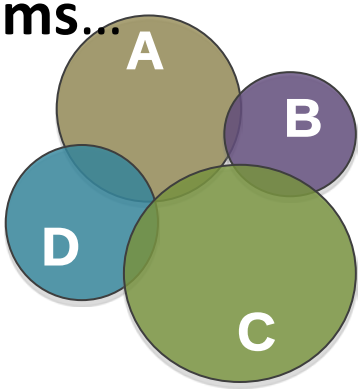
… and more

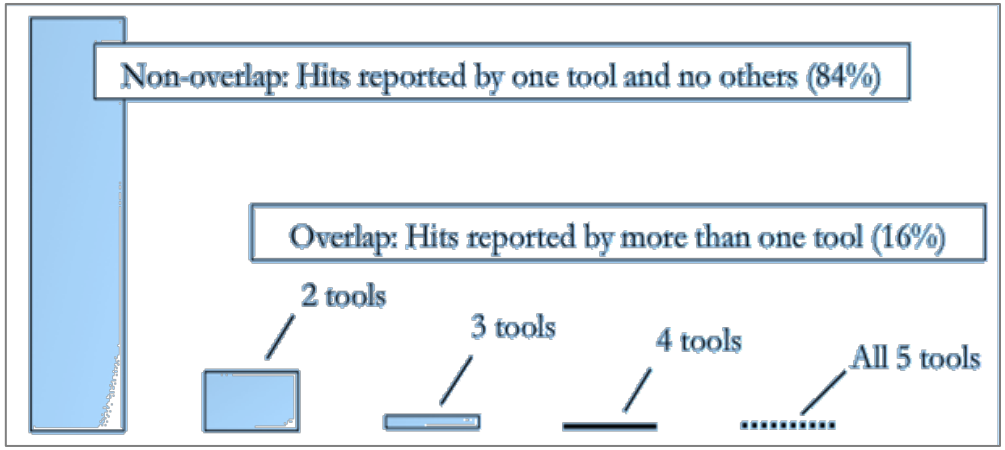
Static analysis tools

Automated analyzers to help you inspect your code

There's no single solution

Different tools identify **different problems**...





Paul E. Black, "Evaluating Static Analysis Tools", 8 July 2009: http://samate.nist.gov/docs/eval_SA_tools_MIT_LL_July_2009.ppt

One static analysis tool on average will only detect

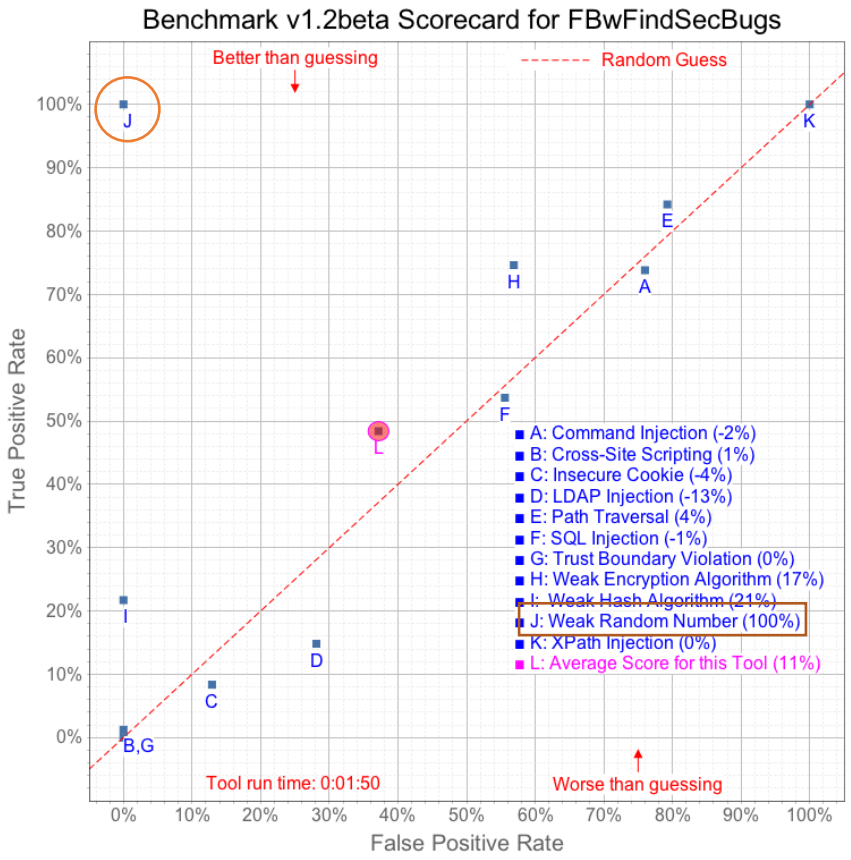
14%

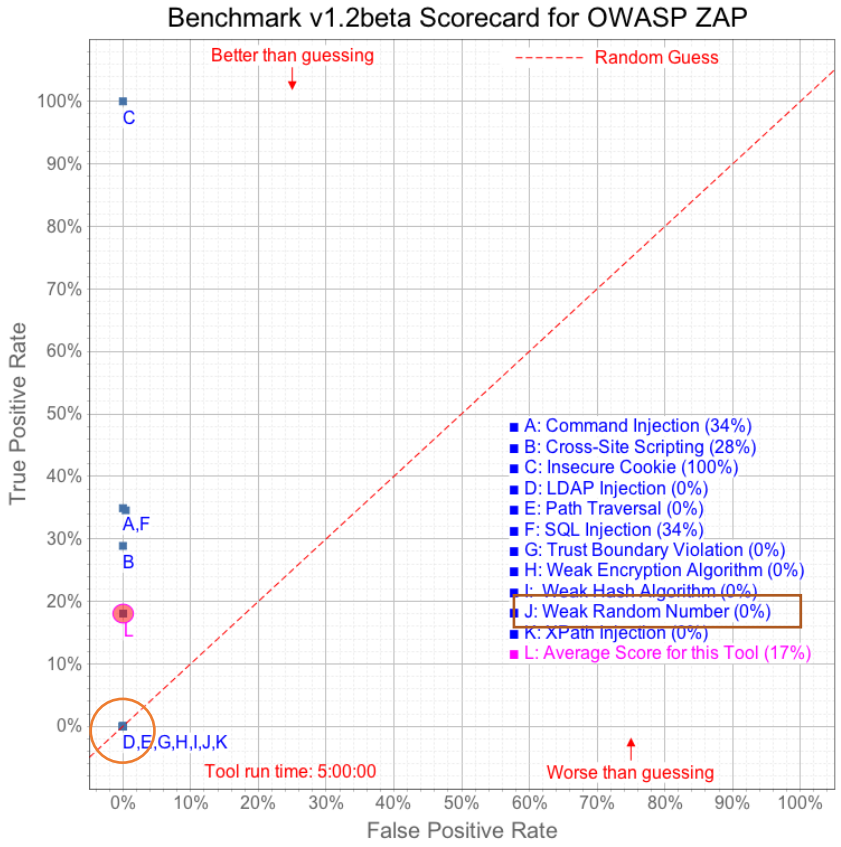
of all weaknesses

Kris Britton and Chuck Willis, "Sticking to the Facts: Scientific Study of Static Analysis Tools", Sept 2011: http://vimeo.com/32421617

> **No tool** stands out as an **uber-tool**.
> Each has its **strengths** and **weaknesses**.
>
> **Kris Britton, Technical Director**
> NSA's Center for Assured Software

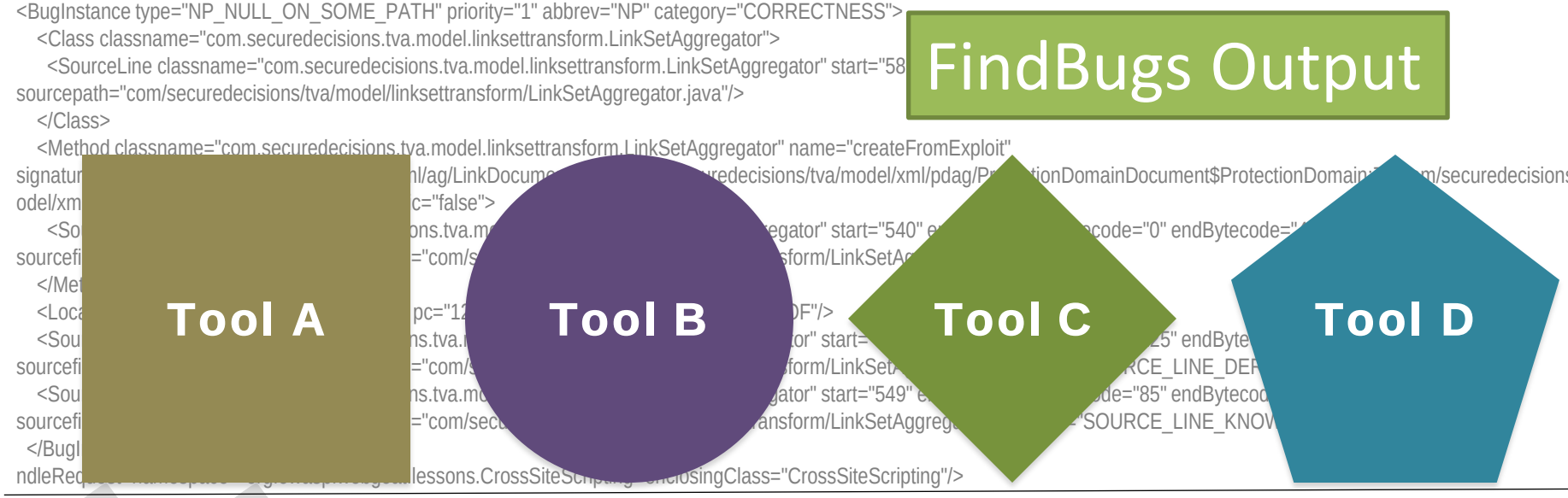
OWASP Benchmark results





https://www.owasp.org/index.php/Benchmark

Tools present their results differently



working with **different tool ve sing and challenging and time-consuming process**: the engines work differently, which is good since they catch different types of problems...

Jim Bird, *Building Real Software*
http://swreflections.blogspot.com/2009_04_01_archive.html

Dynamic analysis tools

OWASP ZAP

AppScan
IBM. Rational.

netsparker®
web application security scanner

VERACODE

acunetix
WEB APPLICATION SECURITY

HP WebInspect™

arachni

How to correlate findings across tools?

https://cwe.mitre.org/

CWE provides a common dictionary of weaknesses which enables correlation across tools

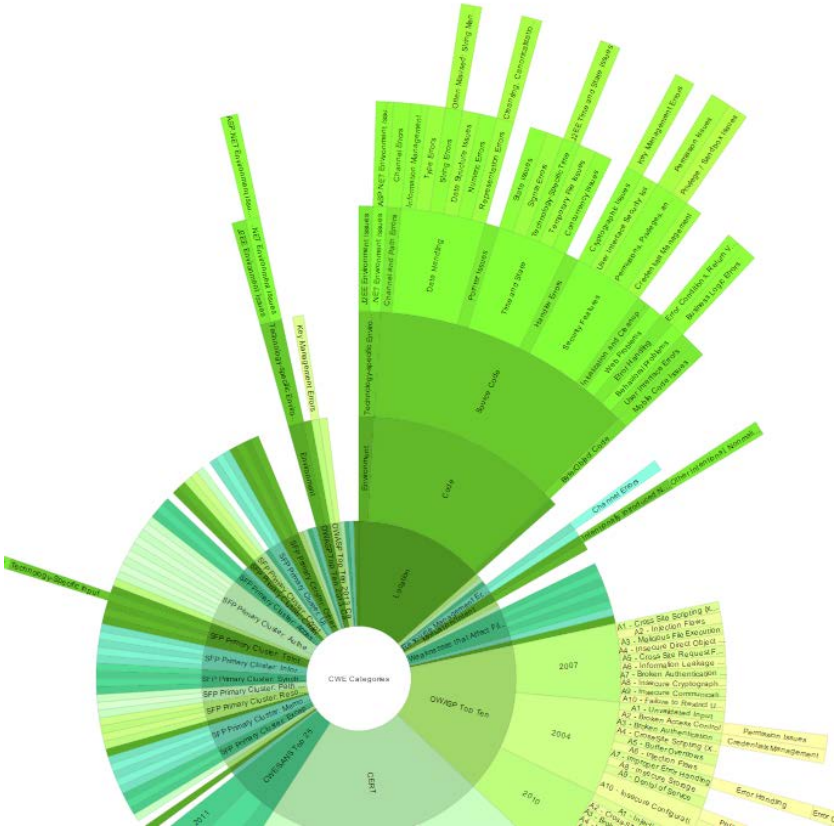
Many commercial tools report a CWE for each finding

… but wait, CWE is not a silver bullet

CWE is complex

CWE is **hierarchical,** and not all vendors have the same thought process on which CWE to use for the same types of issue

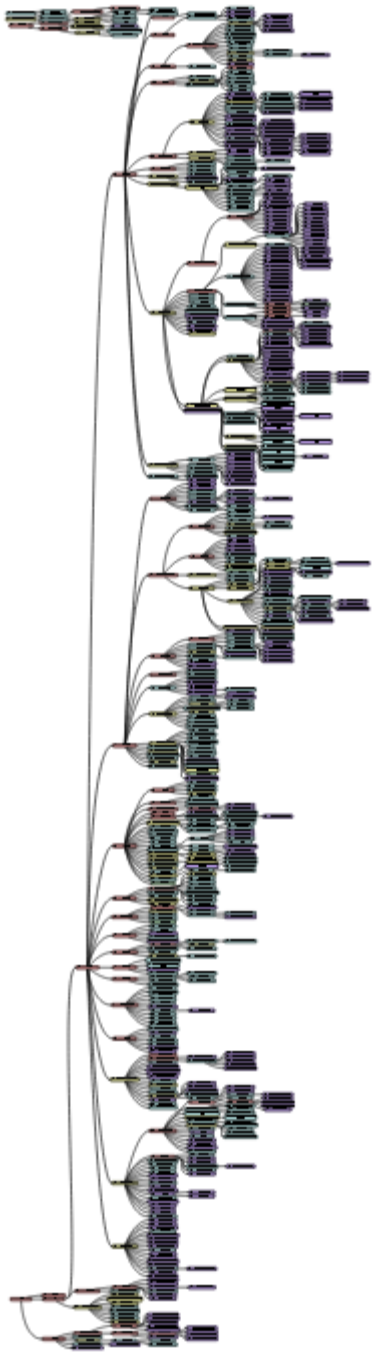
1,557 pages



CWEvis.org

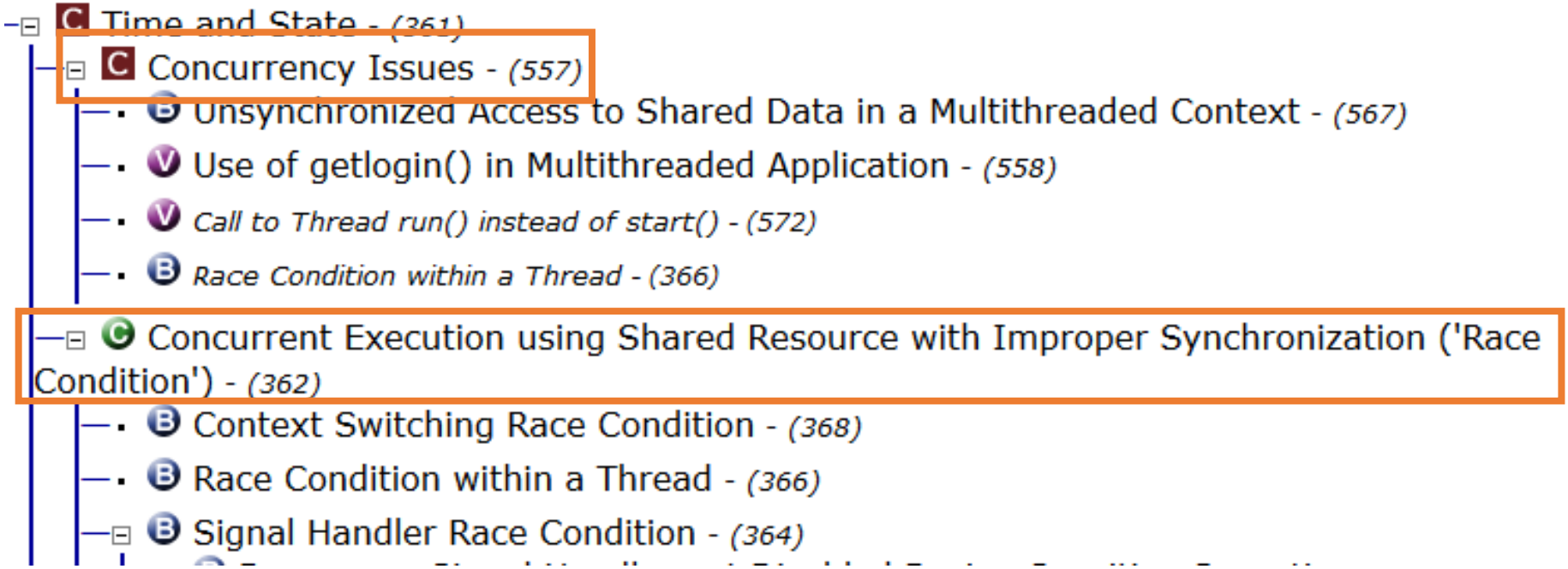
Types

- Views
- Categories
- Weakness Class
- Weakness Base
- Weakness Variants
- Compound Elements
- Chains and Composites

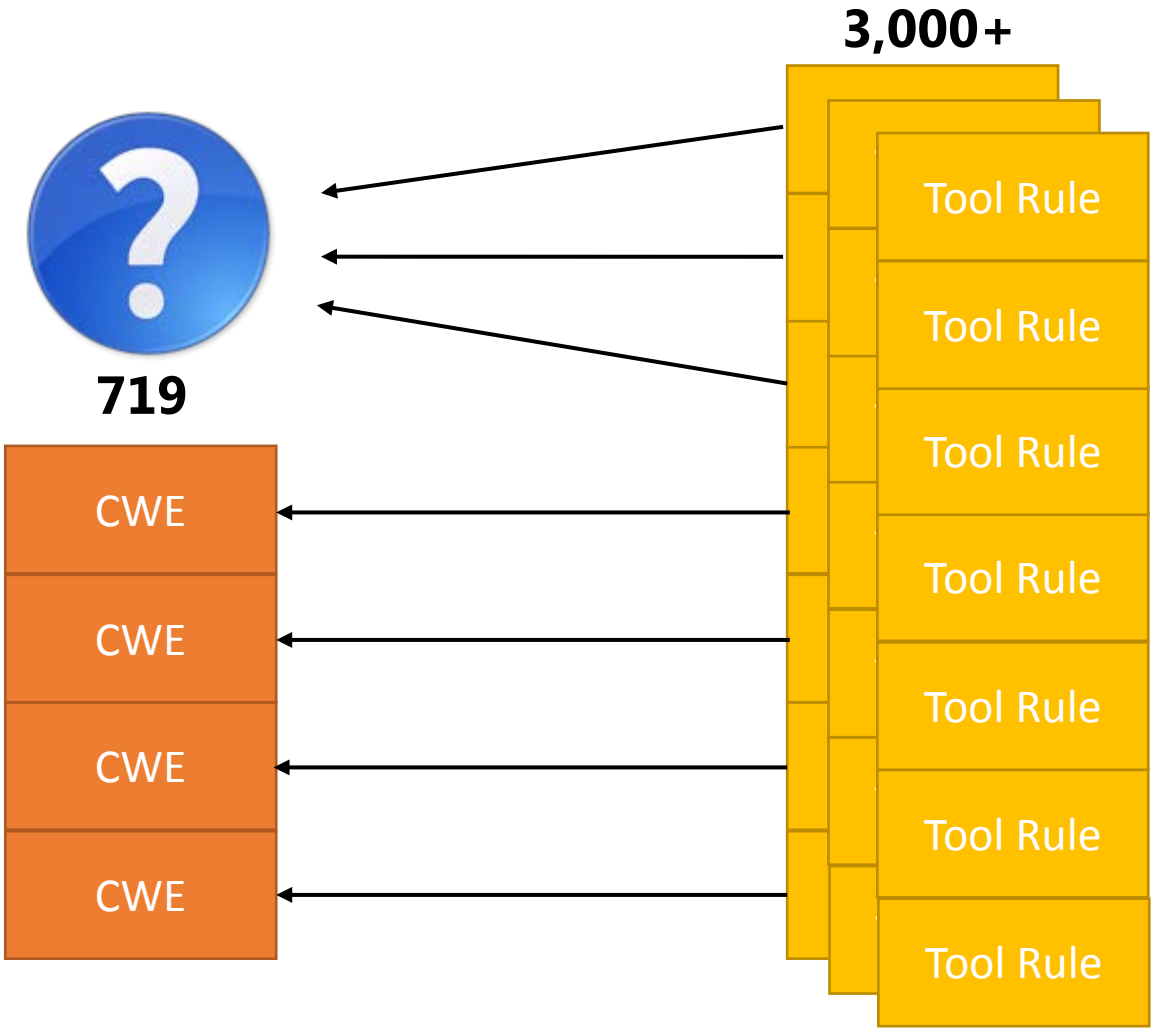


Relationships

- Parent/Child
- CanFollow
- MemberOf
- RequiredBy
- CanAlsoBe


Time and State - (361)
Concurrency Issues - (557)
Unsynchronized Access to Shared Data in a Multithreaded Context - (567)
Use of getlogin() in Multithreaded Application - (558)
Call to Thread run() instead of start() - (572)
Race Condition within a Thread - (366)
Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition') - (362)
Context Switching Race Condition - (368)
Race Condition within a Thread - (366)
Signal Handler Race Condition - (364)

CWE is not always specific enough



Examples…

FindBugs and **PMD** both have a rule for efficiently converting primitives to Strings. No CWE for this:

Example

```
public String convert(int x) {
    String foo = new Integer(x).toString(); // this wastes an object

    return Integer.toString(x);             // preferred approach
}
```

PMD: Avoid the use of temporary objects
FindBugs: Method allocates a boxed primitive just to call toString

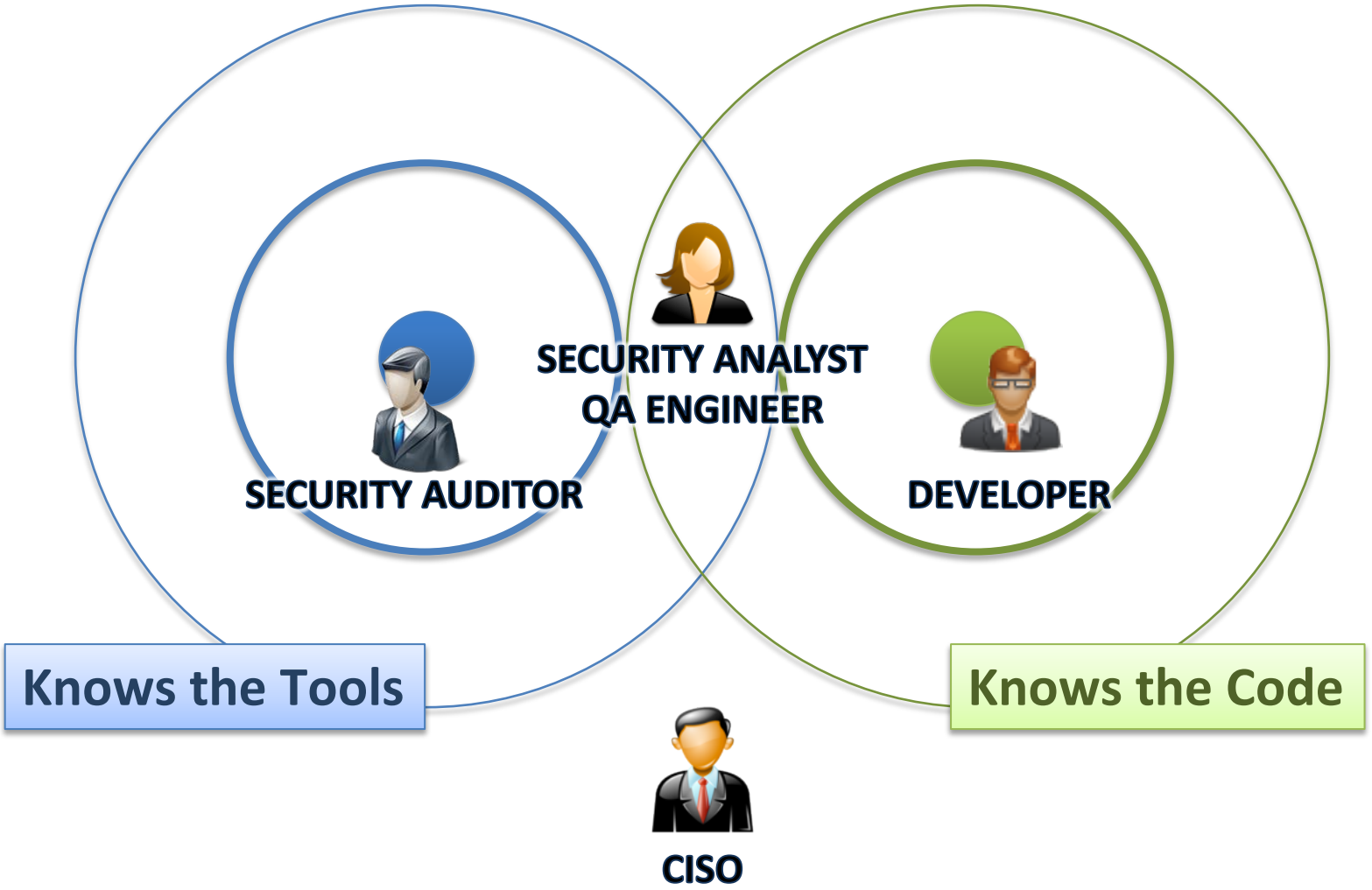
---

Cross-Site Scripting (XSS) – all lumped under CWE-9

- Reflected
- Stored/Persistent XSS
- DOM Based

There is no magic,
you are in total control!

Who does this stuff?



Public resources

CWEvis

CWE Interactive visualization to explore and learn the CWE

cwevis.org

Build Security In

DHS-sponsored educational resource promoting software assurance

buildsecurityin.us-cert.gov

Software Assurance Marketplace (SWAMP)

DHS-sponsored software analysis and testing platform

continuousassurance.org

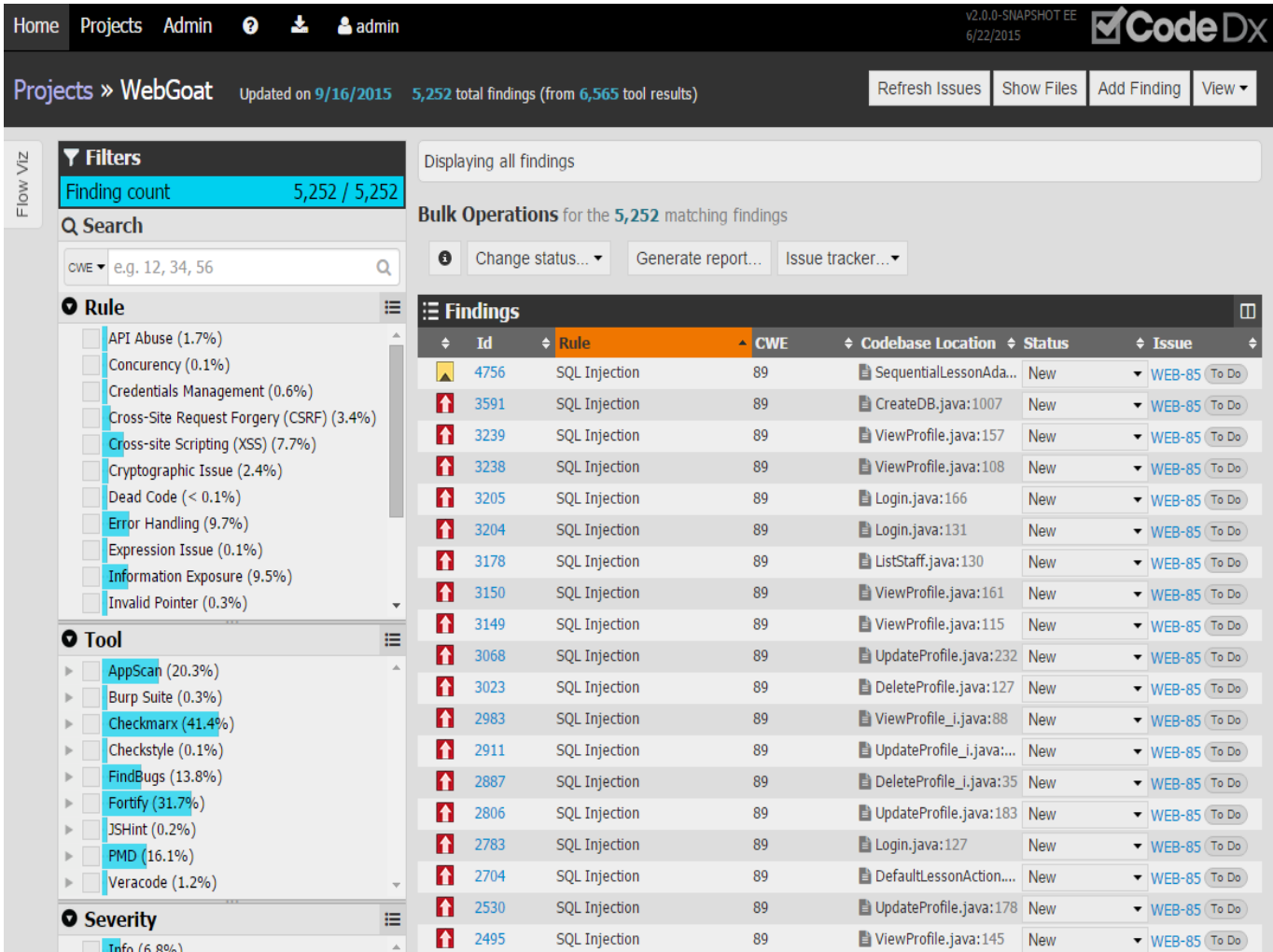
Shameless plug

Code Dx

Software Vulnerability Management

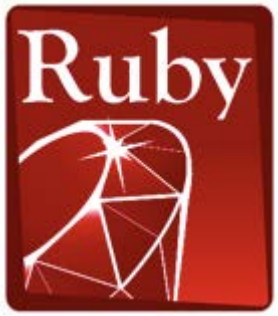
Correlate and explore results from multiple SAST and DAST tools.

Works with open source tools and commercial scanners.



Code Dx supported languages for bundled tools

C/C++



Code Dx Usage Scenarios

MANUAL UPLOAD

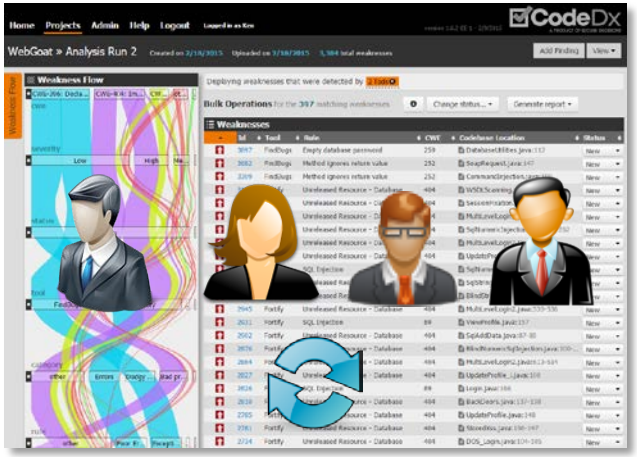
git

Jenkins

REST API

eclipse

Visual Studio



Code Dx

www.codedx.com

eclipse

Visual Studio

Get involved!

www.owasp.org
www.meetup.com/OWASP-Long-Island-Meetup