



OWASP

The Open Web Application Security Project

The Cool Future of Source Code Analysis

**USING THE WISDOM OF THE CROWD TO
ENHANCE APPLICATION SECURITY**

By Moshe Lerner

VP Corporate Strategy at Checkmarx



OWASP

The Open Web Application Security Project

ABOUT MYSELF



Moshe Lerner

VP Corporate Strategy at Checkmarx

- Led many Israeli software companies
- A technology, vision and business match-maker

moshel@checkmarx.com



OWASP

The Open Web Application Security Project

APPLICATION SECURITY ANOMALY

Ratio Developers VS QA experts

2 : 1

Ratio Developers VS Appsec experts

150 : 1

CAN WE TRULY SECURE APPS?



OWASP

The Open Web Application Security Project

NEW PARADIGMS

Agile Development

Continuous Deployment

Beta = GA

Hybrid Apps

CAN WE CREATE A TRUE SECURE SDLC?



OWASP

The Open Web Application Security Project

ISSUES AT HAND

Size | Complexity | Volume

The biggest challenge of current source code analysis solutions is size and agility!

How to deliver:

1. Usable results
2. Automatically
3. Out-of-the-box
4. Accurate

for extra large code bases with thousands+ of results



OWASP

The Open Web Application Security Project

AGENDA

1. How to automatically detect issues that the user does not even know how to describe.
 - “Extracting knowledge” from a large code base
(wisdom of the crowd)
2. How to automatically correlate ???
 - Suggest remediation actions for a fraction of the time for extra large code bases



CHECKMARX



OWASP

The Open Web Application Security Project

ZERO DAYS? ZERO CONFIGURATION?

- What happens if you do not even know what question to ask?
- What if you do not have the resources to configure the system?
- We want a “guru” that asks the questions for us.
 - Configures the system for us.
 - Finds the vulnerabilities for us.
 - Guides us how to fix.

Hold on for a few more slides ...



OWASP

The Open Web Application Security Project

SOURCE CODE ANALYSIS

HISTORY



CHECKMARX



OWASP

The Open Web Application Security Project

FIRST GENERATION CODE ANALYSIS

- The system came out of the box with the relevant security knowledge wired into the system.
- Little-to-no adaptation capabilities.



OWASP

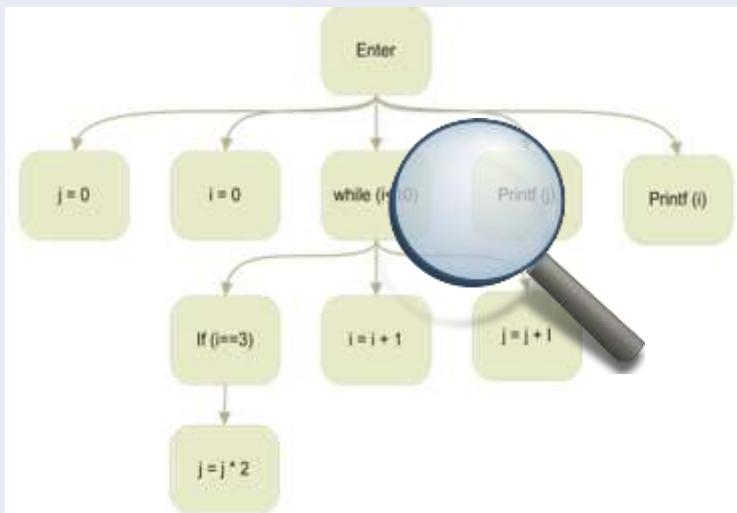
The Open Web Application Security Project

FIRST GENERATION CODE ANALYSIS

```
void main()
{
    int j = 0;
    int i = 0;

    while (i < 10){
        if (i == 3){
            j=j*2;
        }
        j = j + i;
        i = i + 1;
    }

    printf ("%d\n", j);
    printf ("%d,n", i);
}
```





OWASP

The Open Web Application Security Project

NEW GENERATION CODE ANALYSIS

- The system came out of the box with the relevant security knowledge
- Ability to customize existing security knowledge
- Ability to add your own business logic
- EASY!! Virtual Compiler. No need to compile your code.
- EASY!! Incremental scan.
- Detection ranges from SQL Injection to Backdoors



OWASP

The Open Web Application Security Project

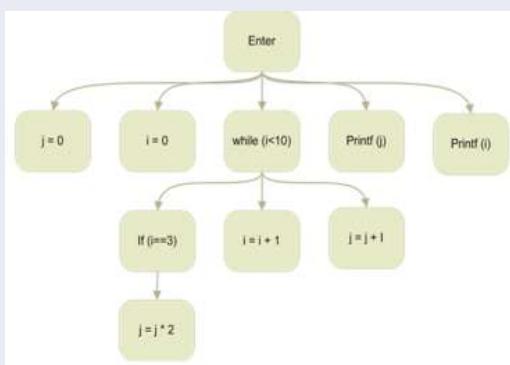
NEW GENERATION CODE ANALYSIS

```
void main()
{
    int j = 0;
    int i = 0;

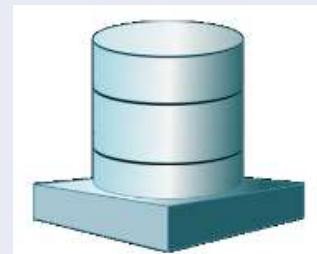
    while (i < 10){
        if (i == 3){
            j=j*2;
        }
        j = j + i;
        i = i + 1;
    }

    printf ("%d\n", j);
    printf ("%d\n", i);
}
```

Abstract



Store



<EXAMPLE/>



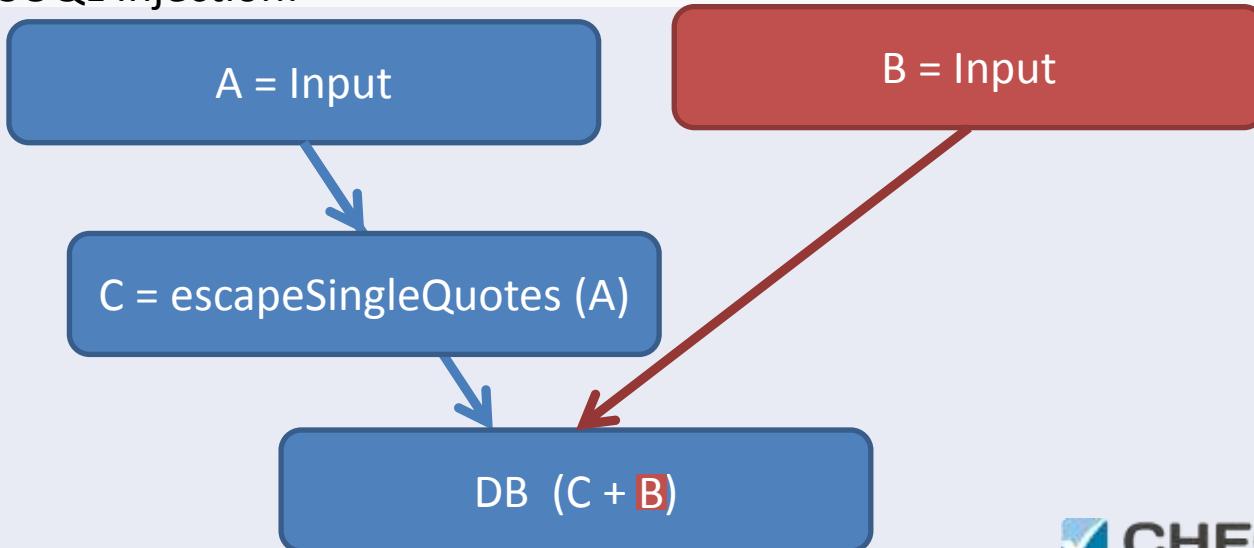
OWASP

The Open Web Application Security Project

SOQL/SOSL Injection

CWE ID	10502
Description	<p>In other programming languages, this flaw is known as SQL injection. Apex does not use SQL, but uses its own database query language, SOQL. SOQL is much simpler and more limited in functionality than for SQL injection, but the SQL/SOQL injection involves taking advantage of this simplicity. If the input is not validated, it is possible to craft an SQL query and trick the application into executing it.</p> <pre>CxList db = Find_DB(); CxList inputs = Find_Interactive_Inputs(); CxList sanitized = Sanitize(); result = db.InfluencedByAndNotSanitized(inputs, sanitized);</pre>

SOQL Injection:



<EXAMPLE/>



```
CxList Input = All.FindByName("input");
```

```
CxList DB = All.FindByName("execute");
```

```
CxList Fix = All.FindByName("fix");
```

```
Return DB.InfluencedByAndNotSanitized(input, fix);
```



OWASP

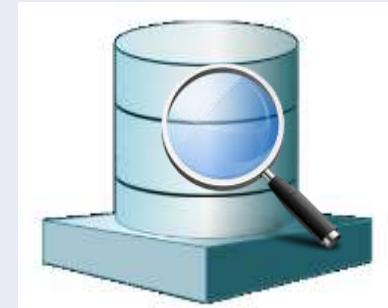
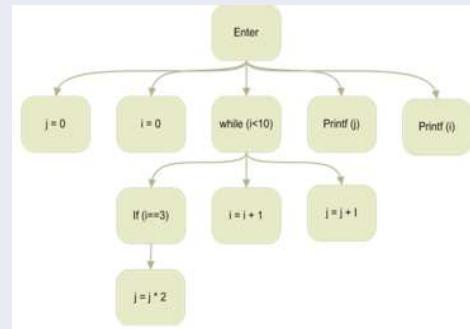
The Open Web Application Security Project

APPLICATION INTELLIGENCE

```
void main()
{
    int j = 0;
    int i = 0;

    while (i < 10){
        if (i == 3){
            j=j*2;
        }
        j = j + i;
        i = i + 1;
    }

    printf ("%d\n", j);
    printf ("%d,n", i)
}
```





OWASP

The Open Web Application Security Project

The Cool Future of Source Code Analysis

SCKD

Source Code Knowledge Discovery

“Using Wisdom of the crowd (Big Data) to identify security vulnerabilities via code irregularities”



OWASP

The Open Web Application Security Project

ZERO DAYS? ZERO CONFIGURATION?

- What happens if you do not even know what question to ask?
- What if you do not have the resources to configure the system?
- We want a “Guru” that asks the questions for us.
- Configures the system for us.
- Finds the vulnerabilities for us.
- Guides us.



CHECKMARX



OWASP

The Open Web Application Security Project

THERE IS SUCH A GURU

YOU

YOU

YOU

and... YOU!

All of you – Wisdom of the crowd

Most of the developers write good, standard, quality code, most of the time



OWASP

The Open Web Application Security Project

CROWD

We can set a baseline based on code statistics and find deviations thereof





OWASP

The Open Web Application Security Project

SCKD

- Source Code Knowledge Discovery – an active research
(Knowledge Discovery in DB - http://en.wikipedia.org/wiki/Knowledge_extraction)

“Knowledge discovery describes the process of automatically searching large volumes of data for patterns that can be considered knowledge about the data. It is often described as deriving knowledge from the input data. Knowledge discovery developed out of the Data mining domain, and is closely related to it both in terms of methodology and terminology.”



OWASP

The Open Web Application Security Project

TECHNIQUE

- Building reference data
- Finding common sequences
- Finding violations



OWASP

The Open Web Application Security Project

BUILDING REFERENCE DATA

```
H = input();  
If (isValid(H))  
{  
...  
    response.write(H);  
...  
}
```

```
A = input();  
If (isValid(A))  
{  
...  
    response.write(A);  
...  
}
```



CHECKMARX

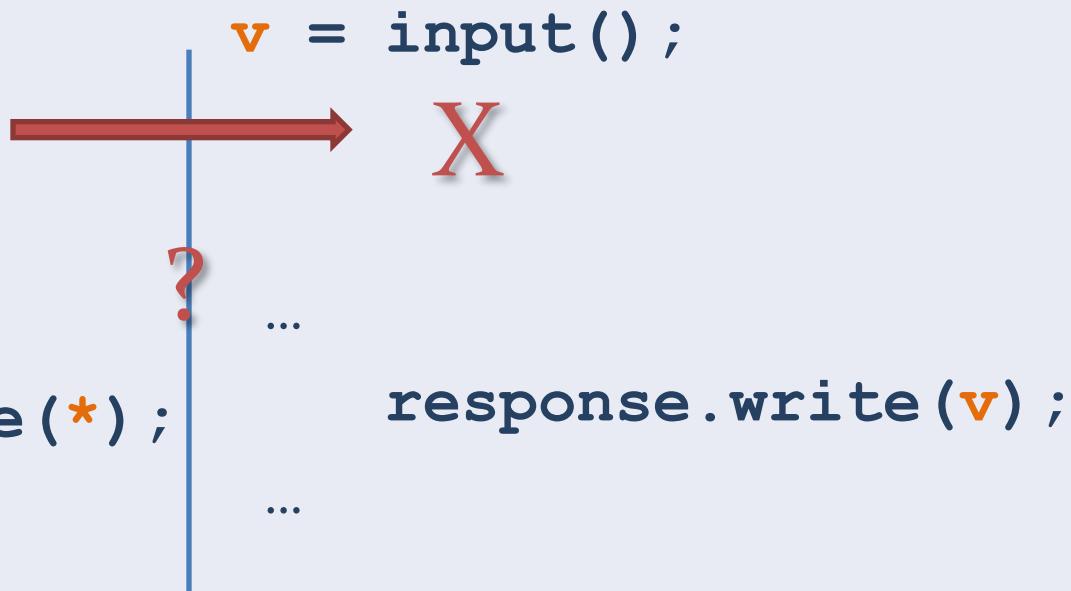


OWASP

The Open Web Application Security Project

FINDING DEVIATIONS

```
* = input();
If (isValid(*))
{
...
    response.write(*);
...
}
```



CHECKMARX



OWASP

The Open Web Application Security Project

BACKDOOR

if my name is Moshe, login

```
If (isAuthenticated(user)) || user.name == "Moshe")
```

```
{
```

```
... ...
```

```
}
```



CHECKMARX



OWASP

The Open Web Application Security Project

EXAMPLE: LEVERAGING CLOUD OF APPS?

Find similarities between
different applications in order to
set an intra-corporate
standard.

With Zero-Definition!

It's enough that some
apps were fixed.
They'll allow us to find
the apps that are not yet fixed.

VAT = 1.05

...

VAT = 1.08

...

VAT = 1.08

...

VAT = 1.08



CHECKMARX



OWASP

The Open Web Application Security Project

WORKS WELL FOR:

- General:
 - We can find the hidden knowledge of the crowd, give it a name and find breaches of it.
- Security:
 - Make sure the user is authenticated at each page
 - Auto-recognize sanitization routines
 - Backdoors (“`if (isValid(user) or user==“Moshe”)...“`)
 - Business logic (“`if (qty > 0) {charge (qty*amnt)}`”)
- Quality
 - Always release a specific resource
 - Best coding practices (auto recognize conventions)
 - Initialize a variable



OWASP

The Open Web Application Security Project

ALSO

- Wisdom of the crowd
- Works better for larger enterprises and code bases



OWASP

The Open Web Application Security Project

GRAPH VISUALIZATION

Optimize call for action

“Using smart graph methods to identify
Vulnerability junctions and best fix locations”



OWASP

The Open Web Application Security Project

ISSUE

- Findings thousands accurate results, does not make us happy ...
- Webgoat, for example, has ~220 XXS+SQL Injection
- Assuming 30 minutes to fix each one + 30 minutes to validate will take 220 hours - ~ 1 month of work
- We'll narrow this down to 16 places
- ~1/14 of the time
- So we have some time to play golf ;)



OWASP

The Open Web Application Security Project

CURRENT SITUATION

Each results has a data flow, presented independently from other findings.



OWASP

The Open Web Application Security Project

SINGLE DATA FLOW PATH- XSS

```
String s = Request.QueryString["param1"];
...
Response.Write(s);
```

```
Request.QueryString["param1"];
```



s



```
Response.Write(s);
```



CHECKMARX



OWASP

The Open Web Application Security Project

ONE IS EASY ... AND 14?!

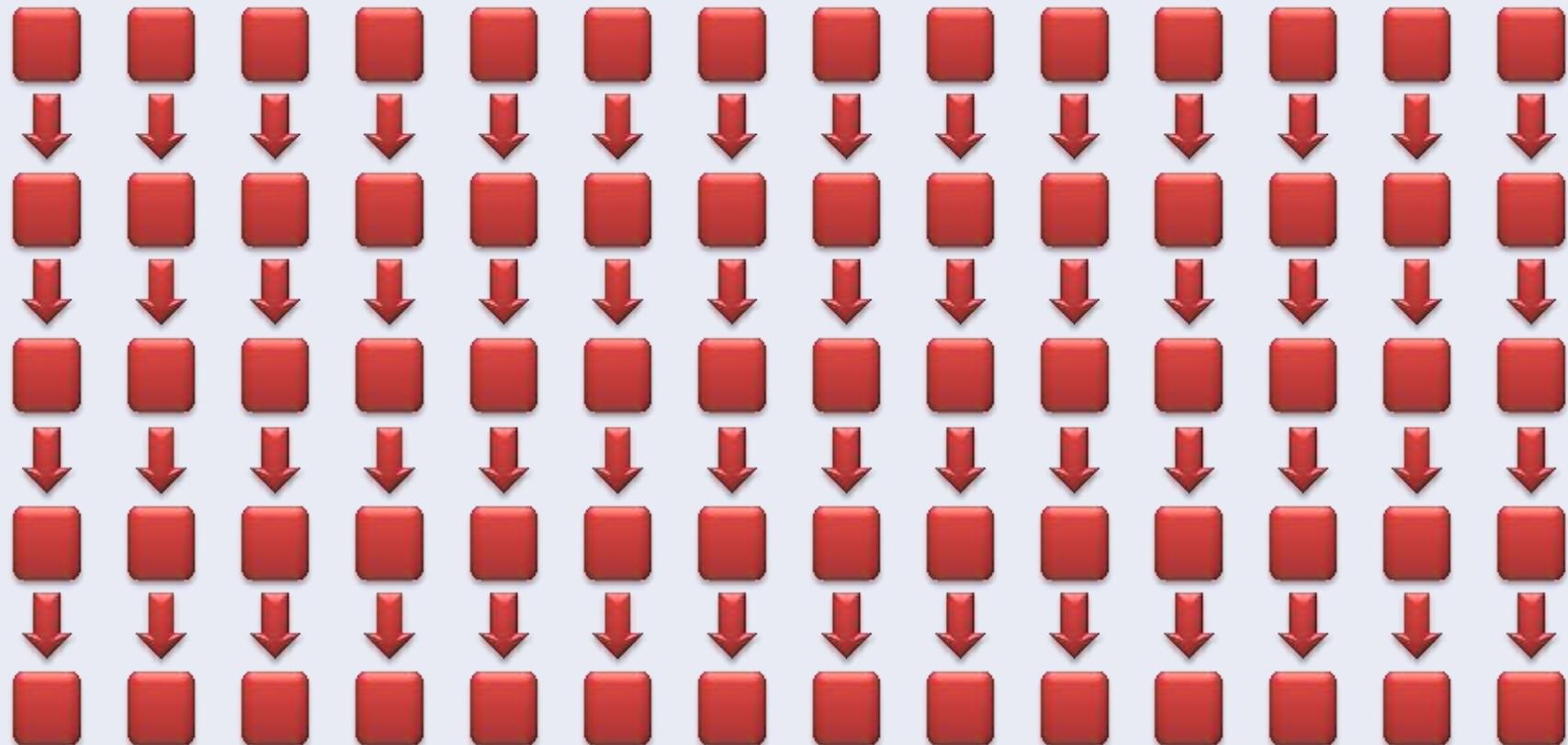




OWASP

The Open Web Application Security Project

Many Single-Path – XSS – a lot of work



CHECKMARX



OWASP

The Open Web Application Security Project

BUT...

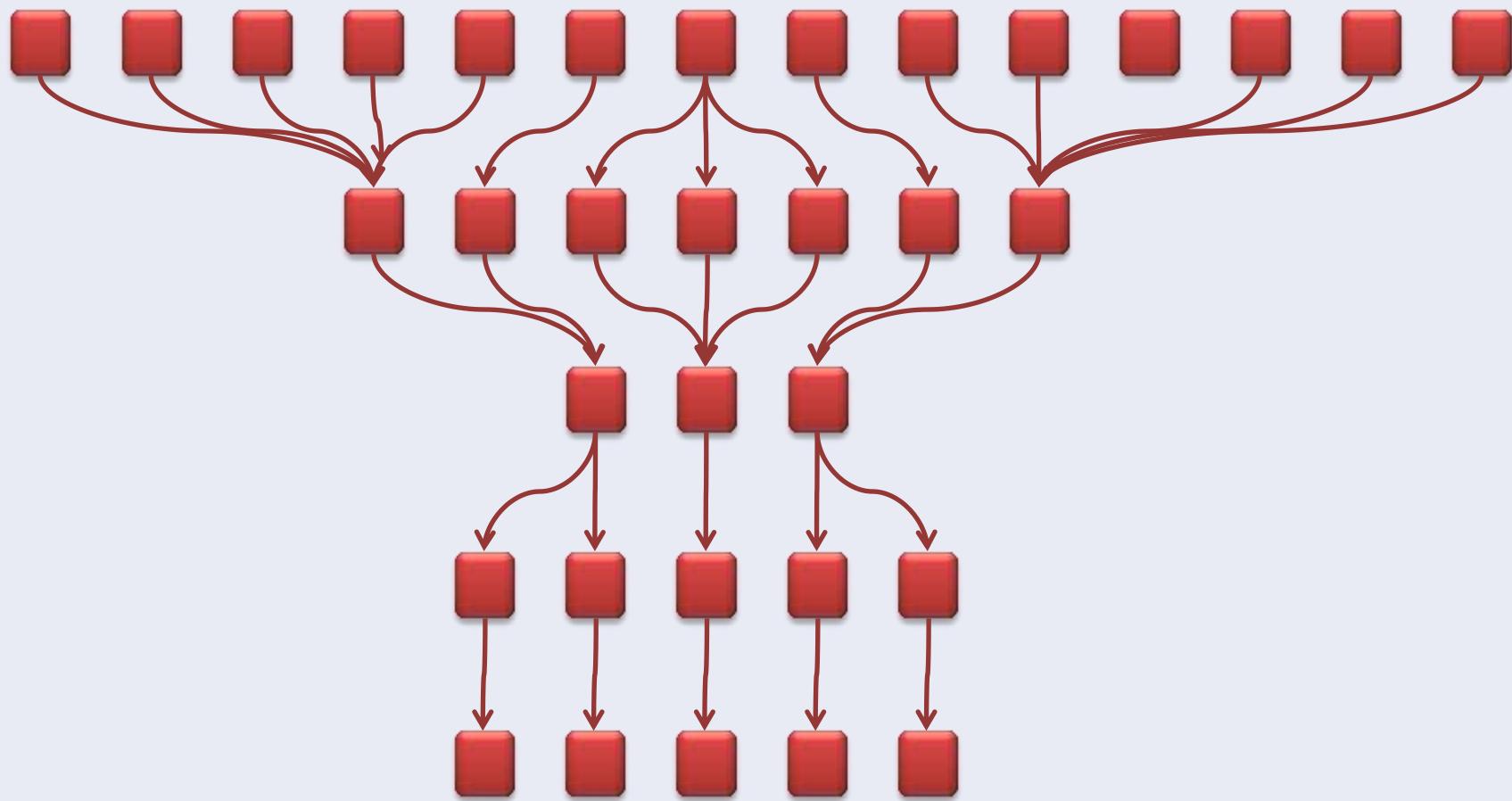
What do they have in common?



OWASP

The Open Web Application Security Project

Combined paths





OWASP

The Open Web Application Security Project

CAN WE:

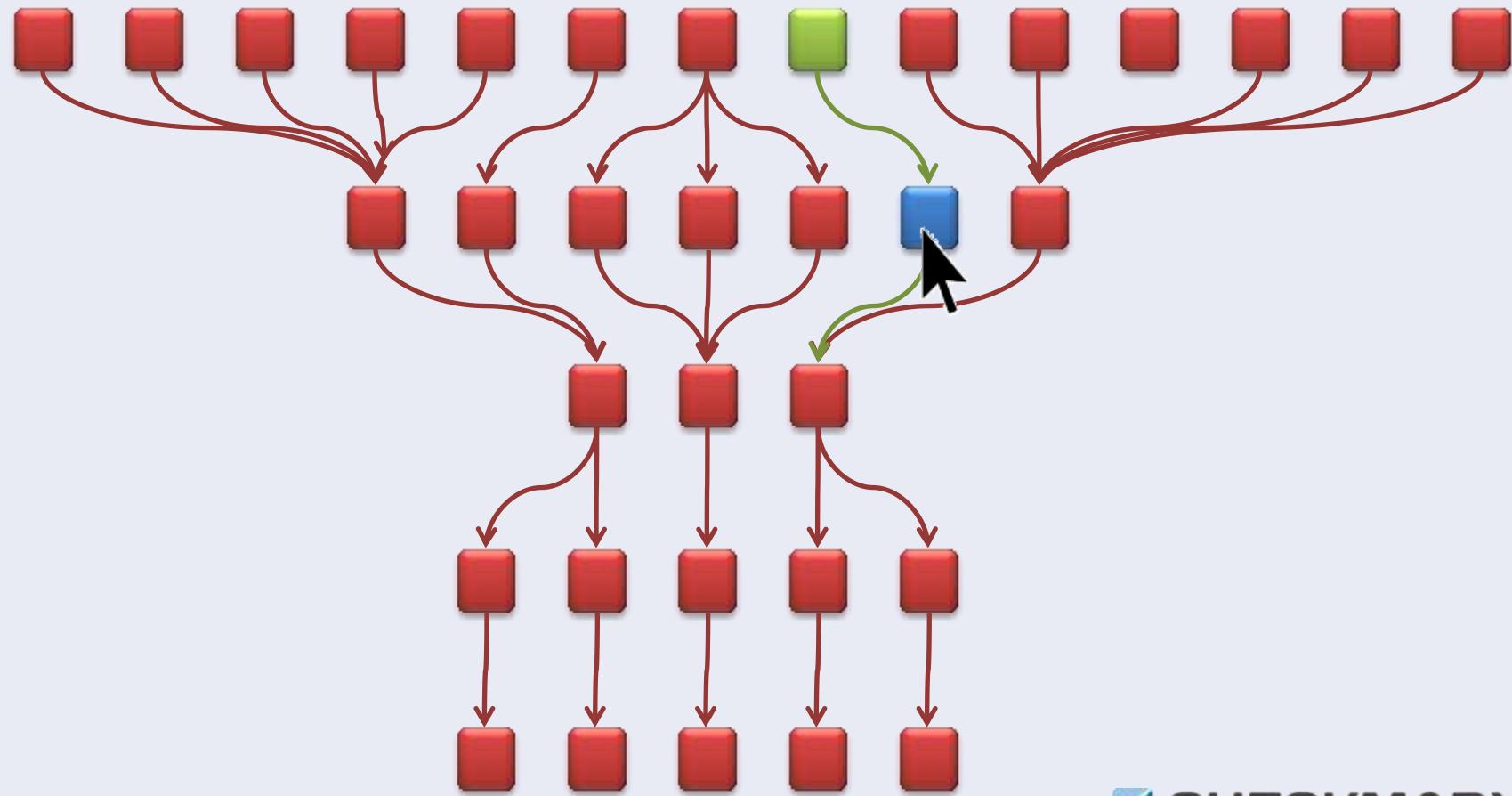
- Point, click and check without even READING the source code?
- “What if I fix here? Or here?”



OWASP

The Open Web Application Security Project

What-If I fix here?

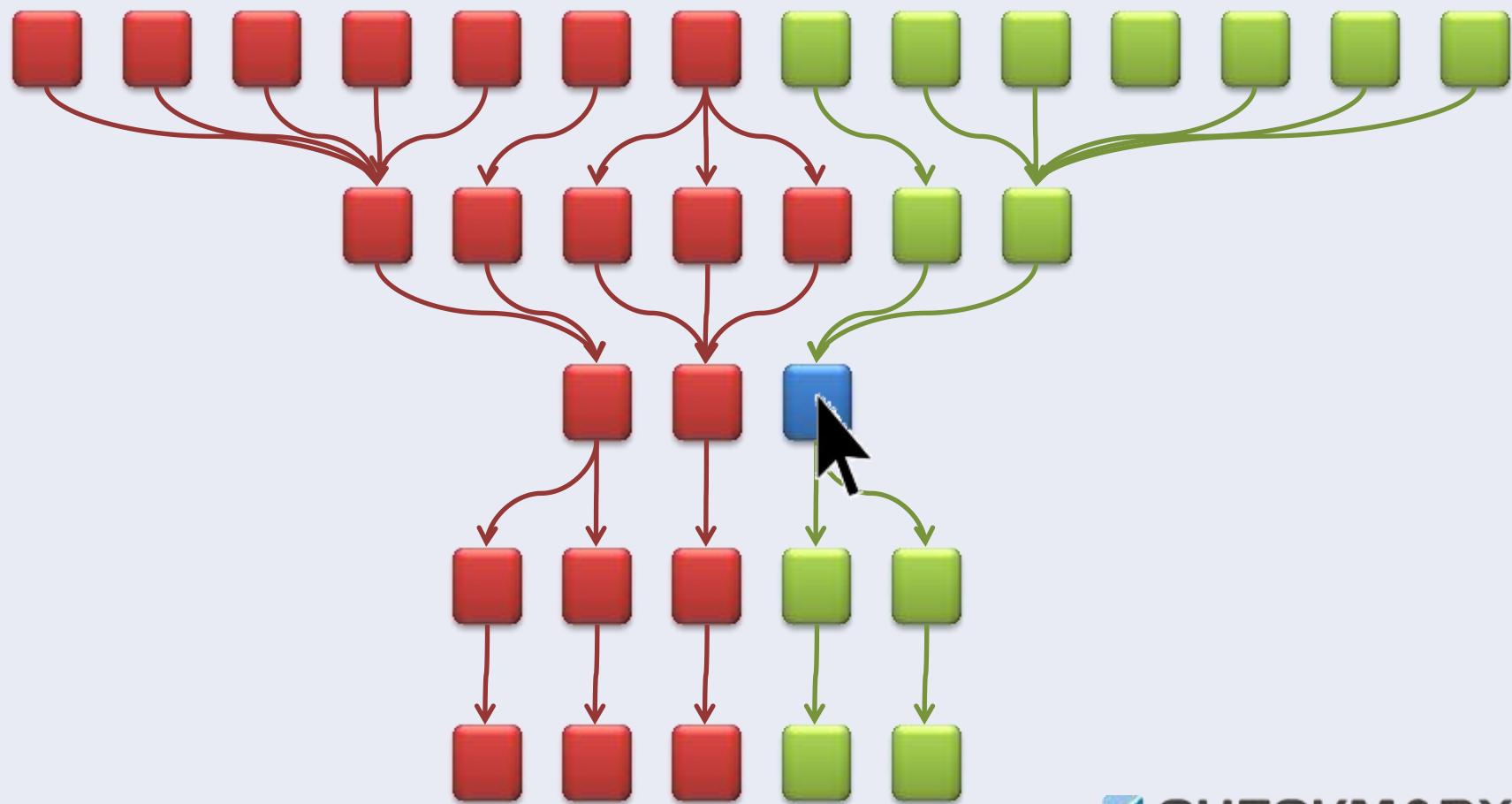




OWASP

The Open Web Application Security Project

Here it is more effective

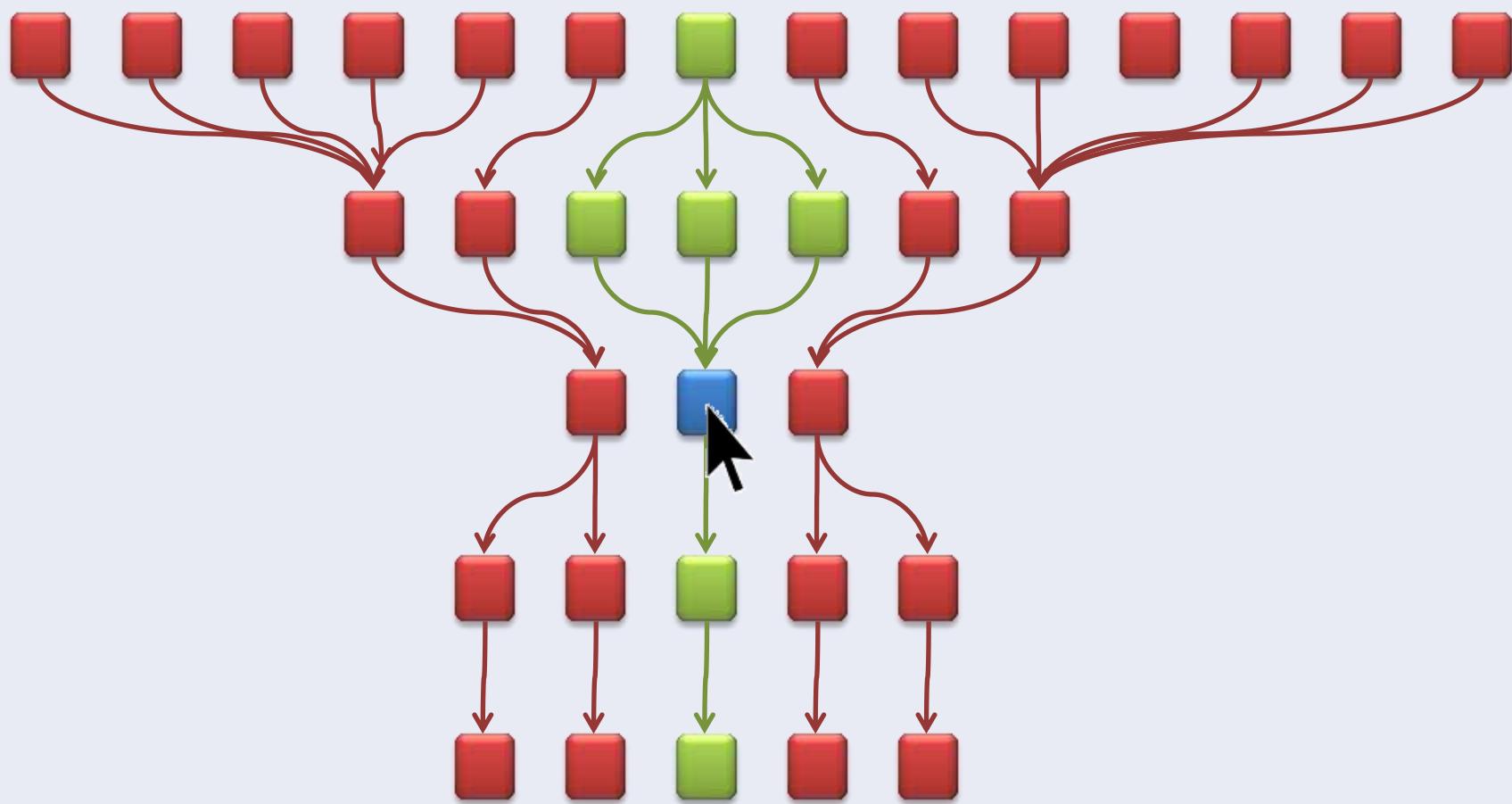




OWASP

The Open Web Application Security Project

And here?

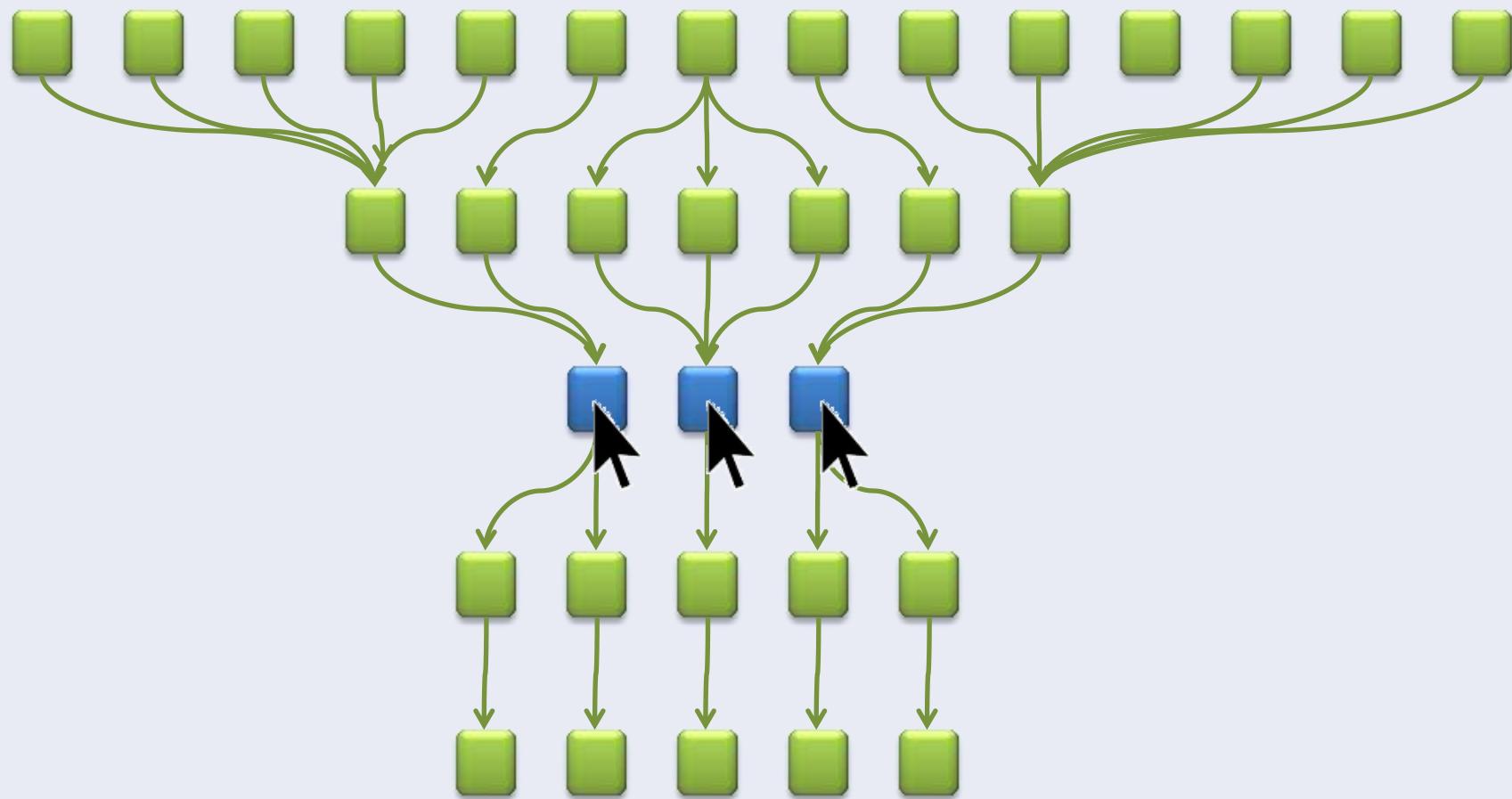




OWASP

The Open Web Application Security Project

Automatic “What-if” => Best Fix Location

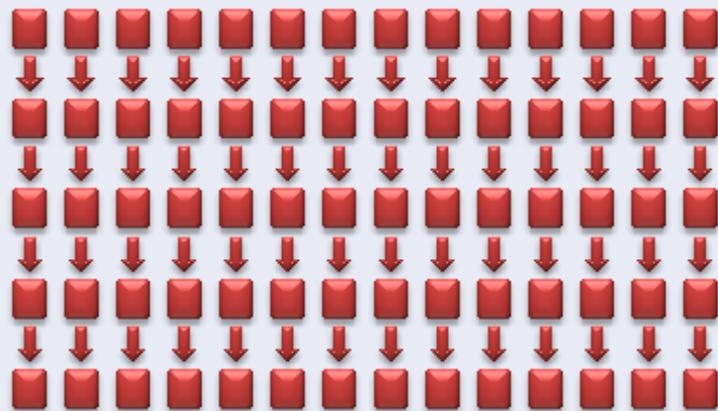




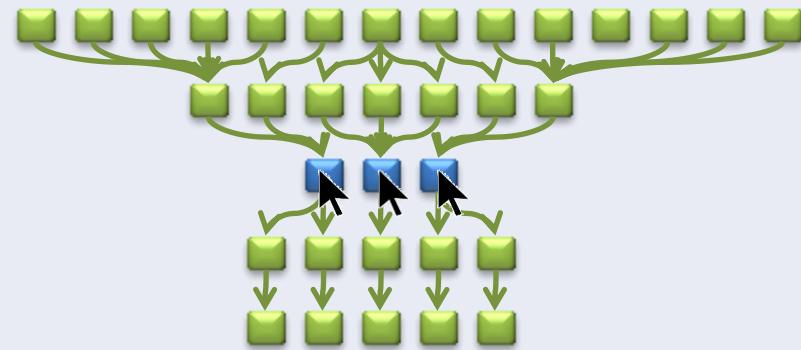
OWASP

The Open Web Application Security Project

Compare the two:



Vs.





OWASP

The Open Web Application Security Project

BENEFITS

- Gives you the correlation between findings of the same type (e.g. SQLi) and different types.
- You are not dealing with individual findings – but with a complete system
- Use your time better



OWASP

The Open Web Application Security Project

FIX LOCATIONS

- At the point of a click we narrow down 220 places into 16.
- The more results, the more effective this solution is



OWASP

The Open Web Application Security Project

RECAP

The biggest challenge of current source code analysis
solutions is size!

How to deliver:

1. Actionable results
2. Automatically
3. Out-of-the-box
4. Accurate

for extra large code bases with thousands+ of results



OWASP

The Open Web Application Security Project

QUESTIONS?



OWASP

The Open Web Application Security Project

Thank you

Moshe Lerner
Moshel@checkmarx.com