



Find and fix software security
problems... wait, do not make
security mistakes in the first place!

SENSEI SECURITY

Faster time to secure code

- Founded in 2016 to create the next generation of solutions to expertly guide developers in writing secure code.

Thought leadership



Experts



Built successful products



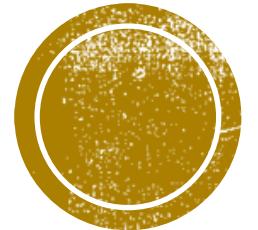
Sensei Security

WHO ARE YOU?

What is your interest in this talk?

- Audience
 - Developer?
 - Security?
 - Students?
- Students, developers, security
- Is Security annoying?
- Are developers annoying?
- Security Courses?





“SECURITY”

What are software security problems?

CREDITS
CREDITS

Sensei Security

SECURITY

What's in the name?

Attack
Hacker
Dropper
Adware
Spoofing
Phishing
Virus
Backdoor
White Hat
service
Malware
Botnet
Cracking
Exploit
Vulnerability
Cybergangs
Malware
Identity
Denial
Attack
Bot
Black
Drive-by
Cybersquatting
Theft
Skimming
Distributed
Flaw
Defacement
Rooting
Sniffing
APT
Cybercriminals
Bug
Hijacker
Cybercrime
Brute-Force



DEVELOPERS INTRODUCE BUGS



Most expensive bugs?

The Ariane 5

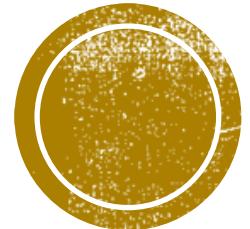
- \$7B development
- \$500M rocket & cargo
- Software Based on the Ariane 4



SOFTWARE SECURITY PROBLEMS

Tell me some application security stories! (SQL injection)





FIND PROBLEMS

Today's view on Application Security: All about finding problems

PEOPLE AND TOOLS

SAST, DAST, IAST, RASP, ...

DAST (Penetration testing solutions and the like)



Peer code review



Static Analysis (SAST)

```
public class Program {
    public Integer next() {
        for(int i = plength - 1; i >= 0;
            i++) if(p[i] > n)
                return p[i];
        else return null;
    }
    throw new NoSuchElementException();
}
```

A snippet of Java code is shown with a magnifying glass focusing on a portion of the code. The code appears to be a part of a linked list iterator. The magnifying glass highlights the condition `i >= 0` and the assignment `p[i] = null`, suggesting a security review for potential buffer overflows or null pointer exceptions.

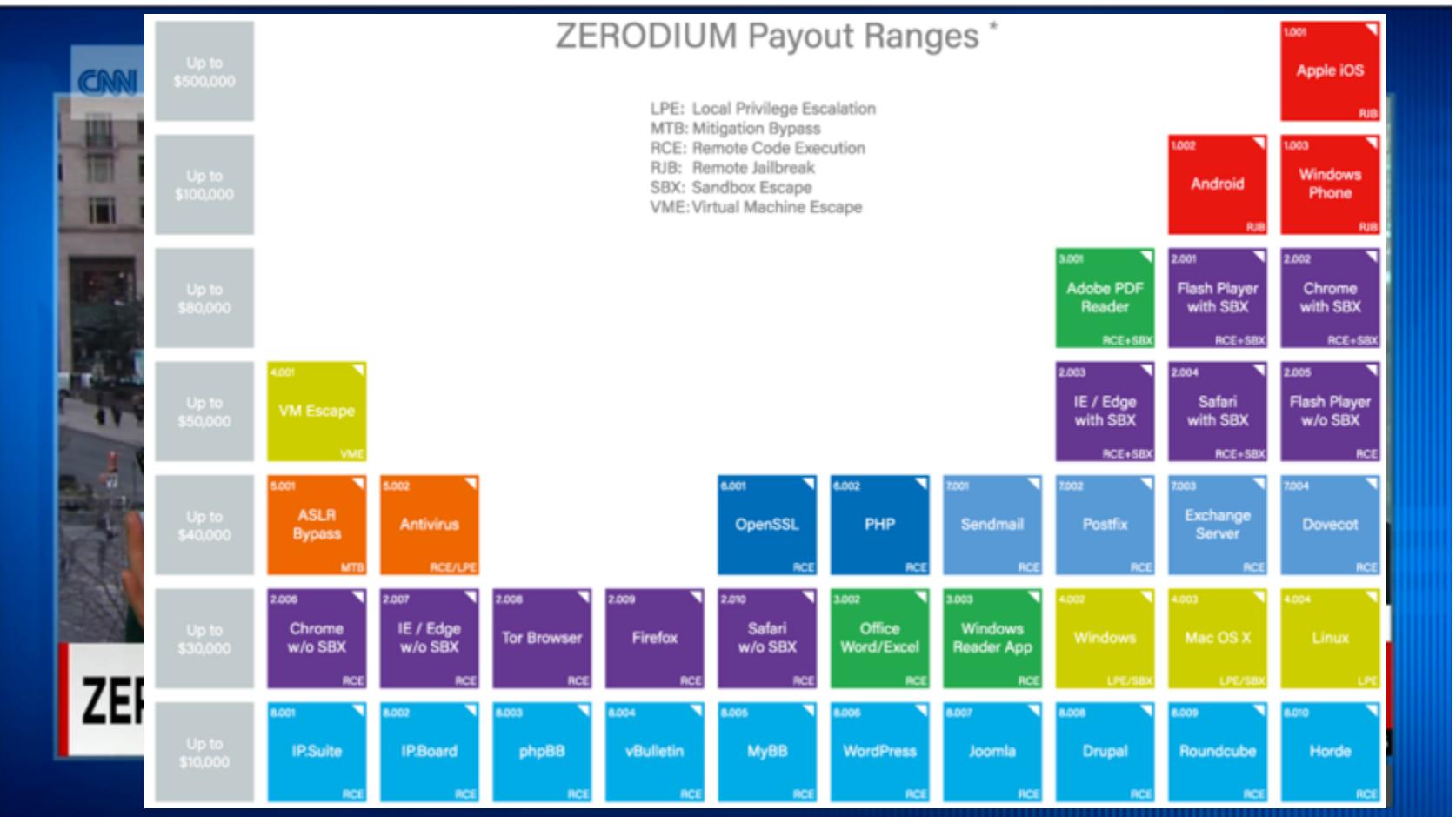
WHITE-HAT HACKERS

Knock knock, who's there?



BLACK-HAT HACKERS

No longer underground



SECURITY IS AWARE OF PROBLEMS

Probably more than 1 problem



Get



Penetration testing solutions



ews



Sensei Security



**FOUND THEM.
NOW FIX THEM!**

Developers vs. Security

CLOUD
SECURITY

Sensei Security

13

WHERE IS THAT PROBLEM?

Fix it!



Sensei Security



Security group



Developers



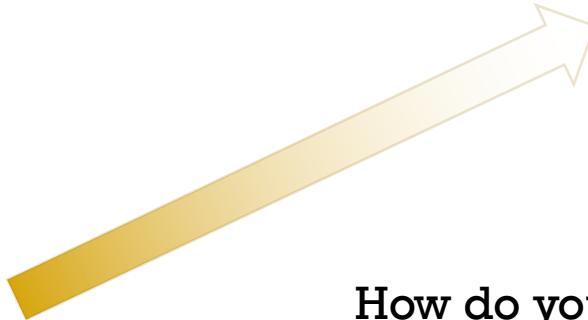
VERACODE
klocwork®



ThreadFix

HOW DO YOU GET THEM FIXED?

Install a process



How do you incentivize developers to get things fixed.

Process in place?



SECURITY PROBLEMS IN SOFTWARE

Great, get me a stool

84%

Of breaches occur at the application layer



(Source: HP Research)

0



Sensei Security

WHO'S HELPING THE DEVELOPER OUT?

Lonely person...

Finding problems (security)



klocwork[®]
AppScan
IBM Rational

VERACODE

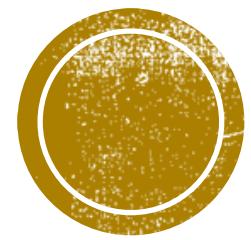


Bug Tracking Systems



ThreadFix





**NOW STOP MAKING NEW
PROBLEMS, PLEASE.**

Approaches

CURRENT
SECURITY



CURRENT
SECURITY

STOP CREATING NEW PROBLEMS

Developers keep on introducing problems.



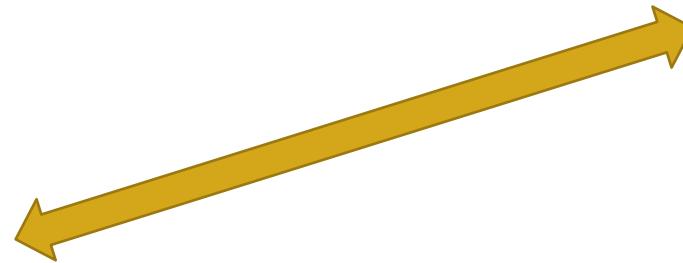
How do they not introduce more problems?

- There are 700 different categories of problems developers can make!
- Detection happens fairly late.

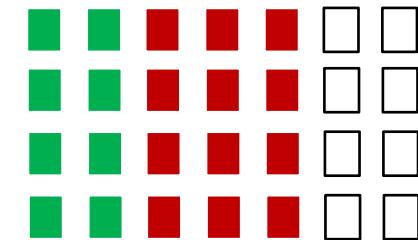


DEVELOPERS: FEATURES

Developers are paid to develop features, not to learn about security!



Calendar



Build features

Fix bugs, security training, ...



HOW DO WE FIX UNKNOWN ISSUES

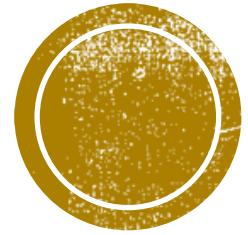
How can we fix what we do not see?



When fixing all the known problems, is your code secure?

Fixing is not robust and consistent programming





**ANY OTHER SOLUTIONS TO
THIS PROBLEM?**

CORE
CLOUD

Sensei Security

REMEMBER . . .

Lonely developer, no help

Finding problems (security)



AppScan
IBM Rational

klocwork®

VERACODE



Sensei Security

Bug Tracking Systems



Bugzilla



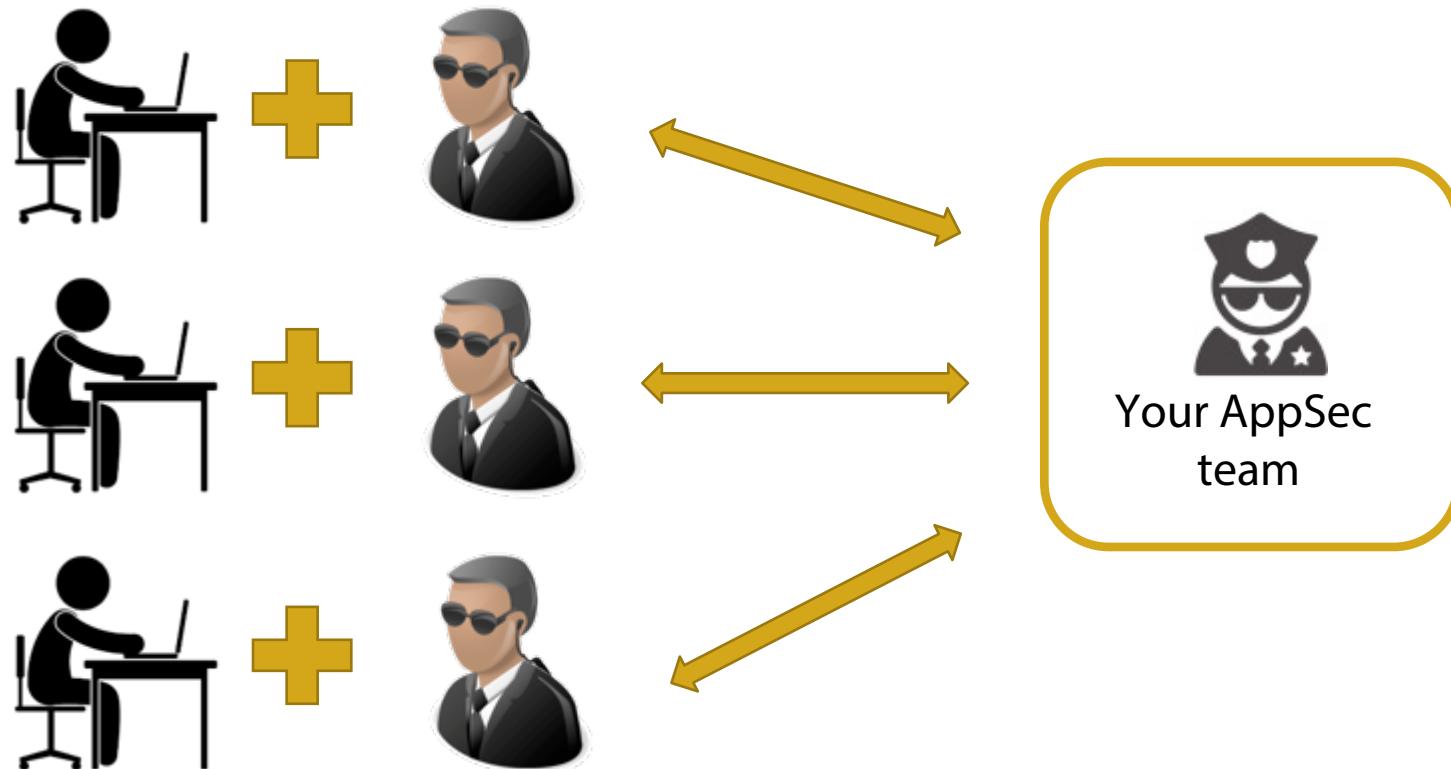
Developer has to fix



ThreadFix

HELP THE DEVELOPER

A software security person next to every developer?



FIND SECURITY PEOPLE

Talent shortage in security



ant in 2014



in 2020 in IT Security



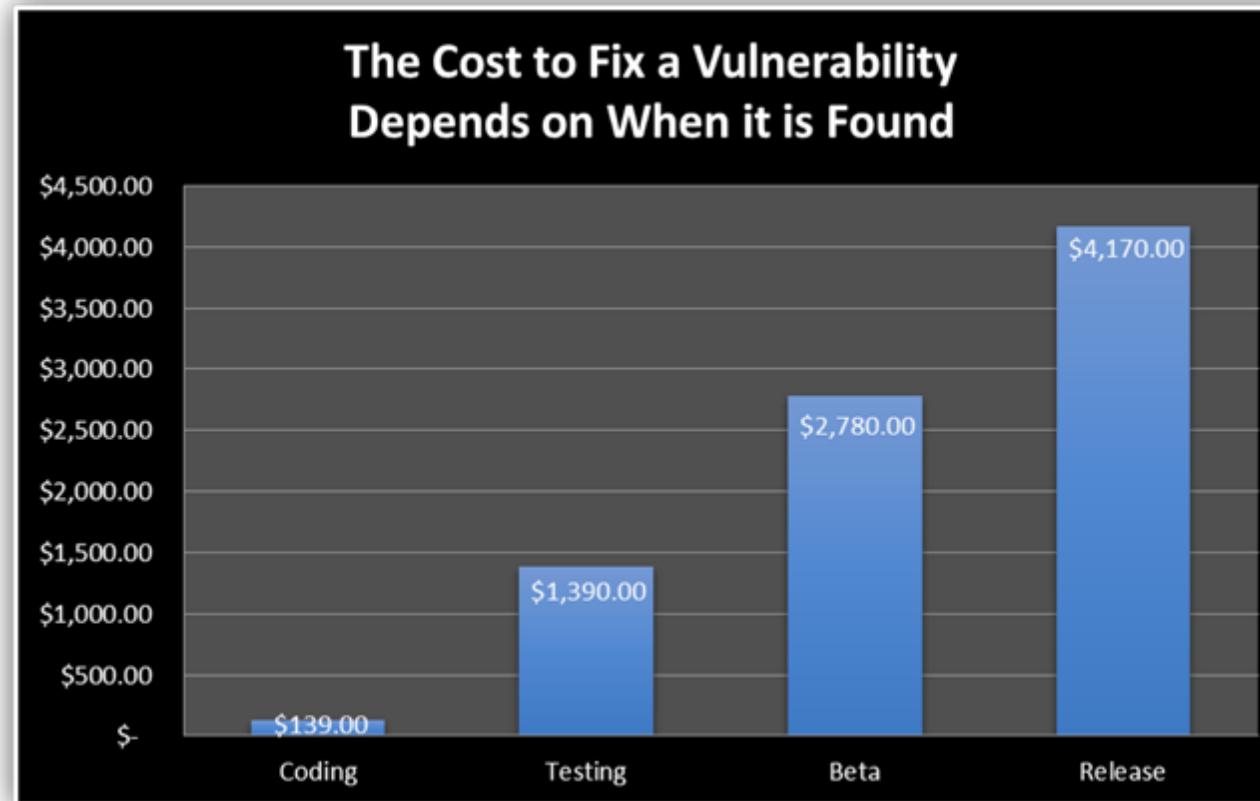
members/100 developers.



Sensei Security

BENEFITS FINDING PROBLEMS EARLY

Cost to fix vulnerabilities



COMPLIANCE REASONS, STANDARDS

Moving towards fixing the root cause



PCI DSS requirements



NIST Special Publication 800-53



International
Organization for
Standardization

ISO/IEC 27034



Sensei Security

ADVANTAGES

Be as close as possible to developers



Faster development



Less security bugs



Pro-active and positive



On the job training

MEASURE SOFTWARE SECURITY



SPIDER CHART FOR FAKE FIRM



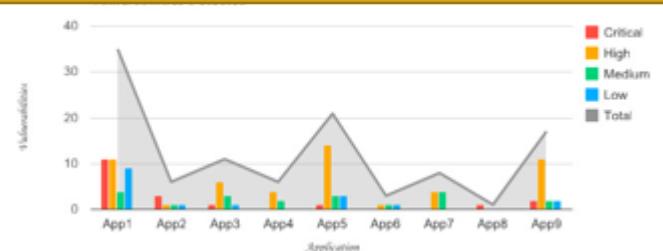
Issues By Priority



Vulnerability Type Distribution



How about: How many problems got we fixed?
How many vulnerabilities did our developers avoid?



THANKS!

Matias Madou Ph.D.



@SenseiSecurity



<https://www.linkedin.com/company/sensei-security>



info@senseisecurity.com



@mmadou



<https://www.linkedin.com/in/matiasmadou>



mmadou@senseisecurity.com



PEOPLE INTERESTED?

We are looking for you!



Benefits:

- Work on cutting edge technology
- Work with cool people
- Want to work when you want
(remote, on-site, day, night,
whenever, wherever)

Come talk to me
or shoot me an e-mail!

jobs@senseisecurity.com
interns@senseisecurity.com



Sensei Security

LOOK FOR FEEDBACK ON OUR PRODUCT

Is your organization serious about application security?



Your organization:

- Want to get software security right in a cost effective way
- Does not want to transform developers into security ninjas (otherwise, keep on training)
- Want to avoid juniors to introduce new problems
- Doesn't matter if you are using point and shoot software security solutions like static analysis solutions, penetration testing, ...
- 50 Java developers.

Come talk to me or shoot me an e-mail!

mmadou@senseisecurity.com



QUESTIONS?

Matias Madou Ph.D.



@SenseiSecurity



<https://www.linkedin.com/company/sensei-security>



info@senseisecurity.com



@mmadou



<https://www.linkedin.com/in/matiasmadou>



mmadou@senseisecurity.com



Sensei Security