

Vulnerability Management – From B Movie to Blockbuster

Rahim Jina

5 December 2018





Rahim Jina

COO & Co-Founder

Edgescan & BCC Risk Advisory



@rahimjina
rahim@edgescan.com





UCLA

Kiplinger



HACKED

AMNES
INTERNATI

Matillion®

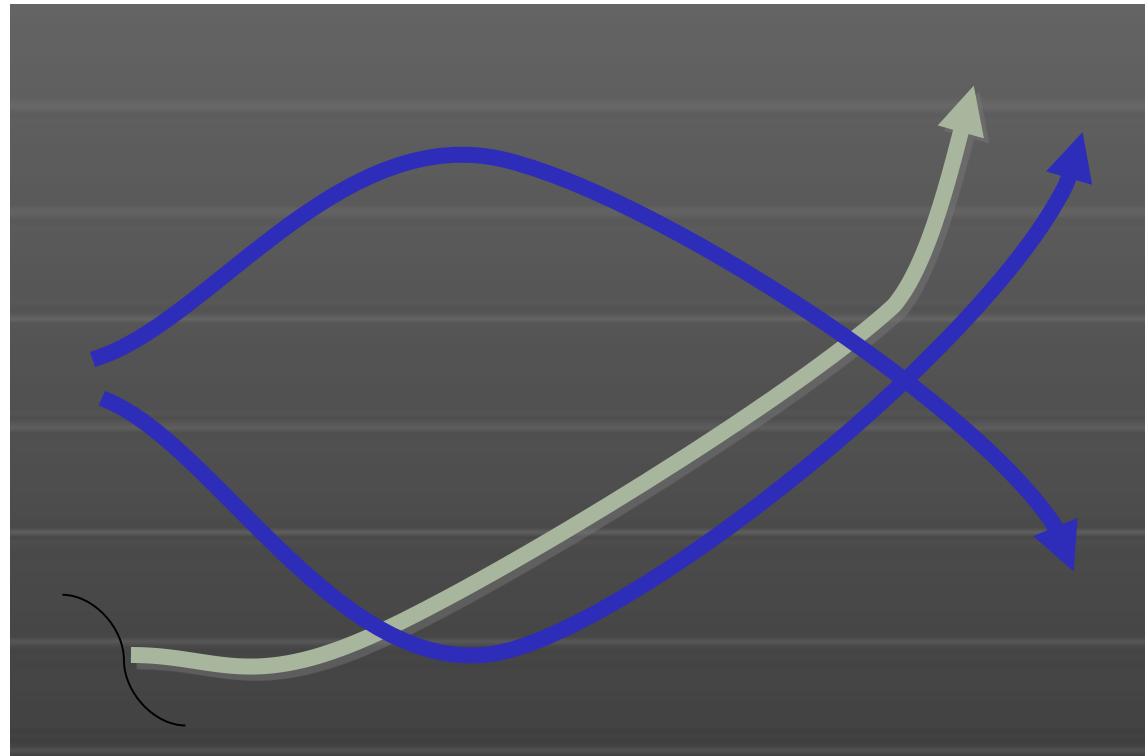
RATFOR
GLOBAL INTELLIGENCE

UNIVERSITY

WOODS



Its (not) the \$\$\$\$



Information
security spend

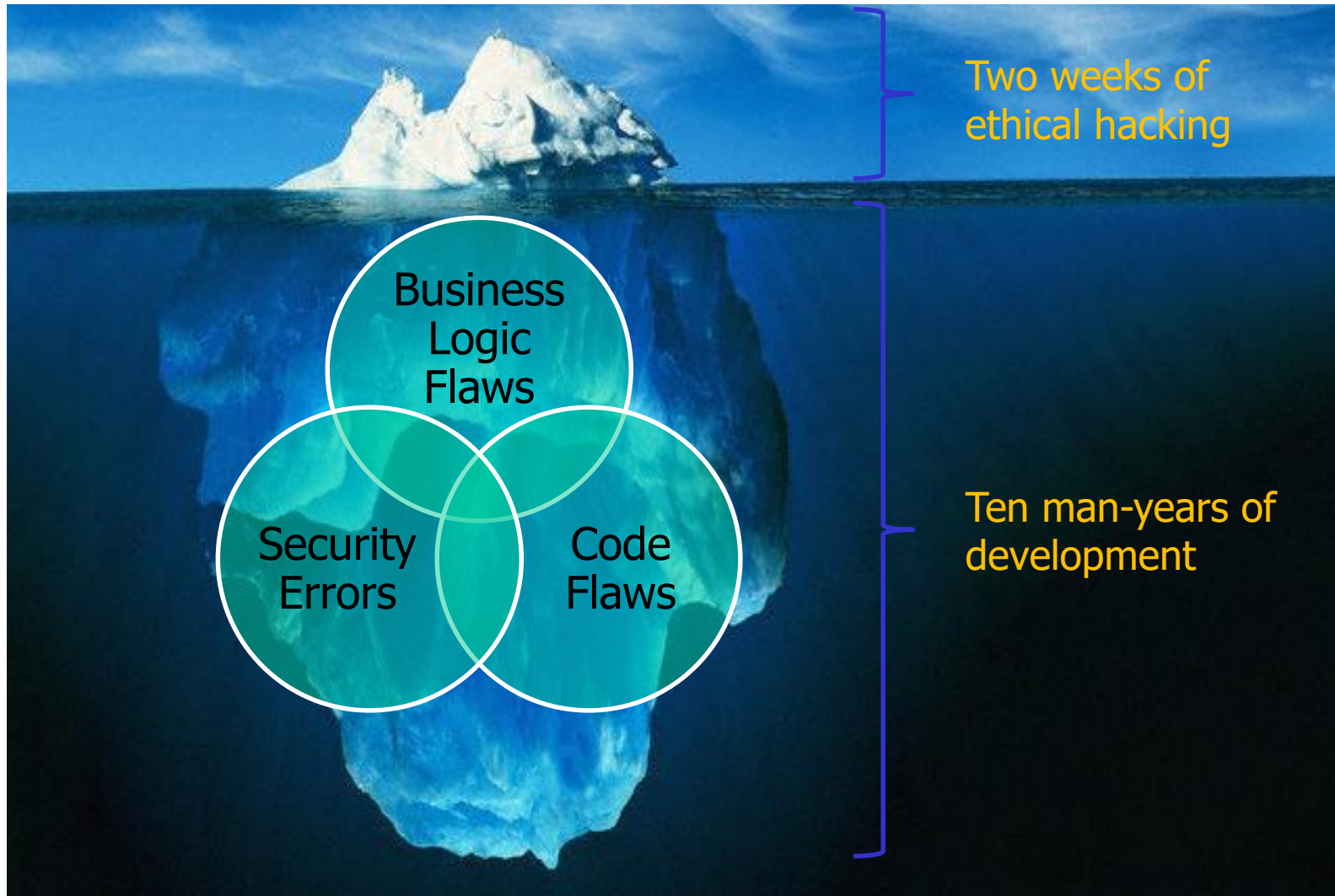
Security incidents
(business impact)

Problem # 1

Asymmetric Arms Race



An inconvenient truth



In two weeks:

Consultant "tune tools"

Use multiple tools – verify issues

Customize Attack Vectors to technology stack

Achieve 80-90 application functionality coverage

How experienced is the consultant?

Was the environment even working properly!!

Are they as good as the bad guys?

They certainly need to be, they only have 2 weeks, right!!?

Code may be pushed to live soon after the test.

Potential window of Exploitation could be until the next pen test.

6 mths, 9 mths, 1 year?

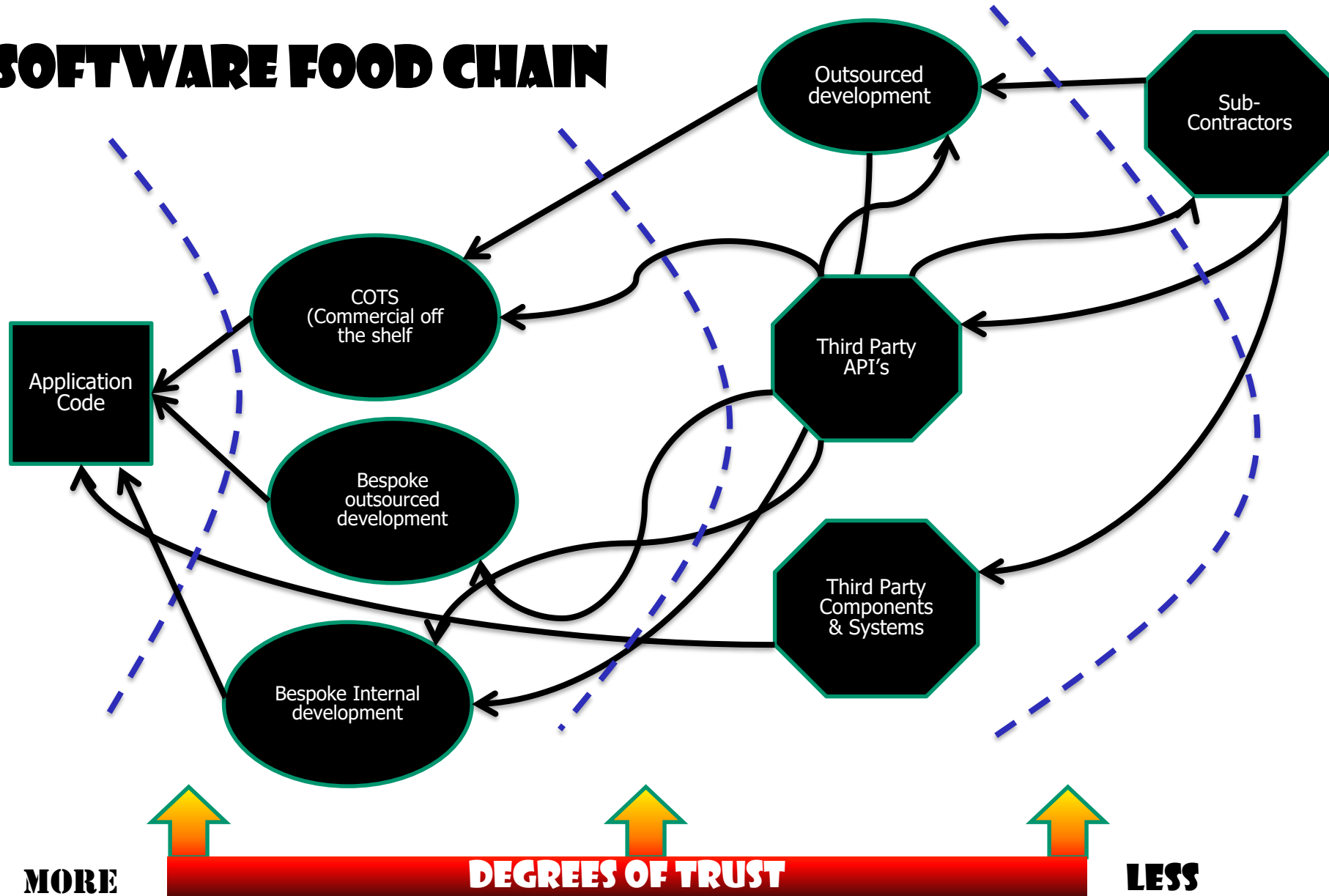




Problem # 2

You are what you eat

SOFTWARE FOOD CHAIN



You may not let some of the people who have developed your code into your offices!!

2018 - Open Source Security Statistics.

- 23% of the Components in the Average Software Application Contain Known Vulnerabilities
- 60% of businesses do not keep a complete inventory (bill of materials) of components being used in their applications.

Struts - application development framework
: downloaded 2 million times in the last
year. –

Remote Code Execution attack CVE-2017-
9805

Struts 2.1.2 - 2.3.33, 2.5 - 2.5.12

<https://cwiki.apache.org/confluence/display/WW/S2-052>

2.1.2 – 9 years old

2.3.33 – July 2017

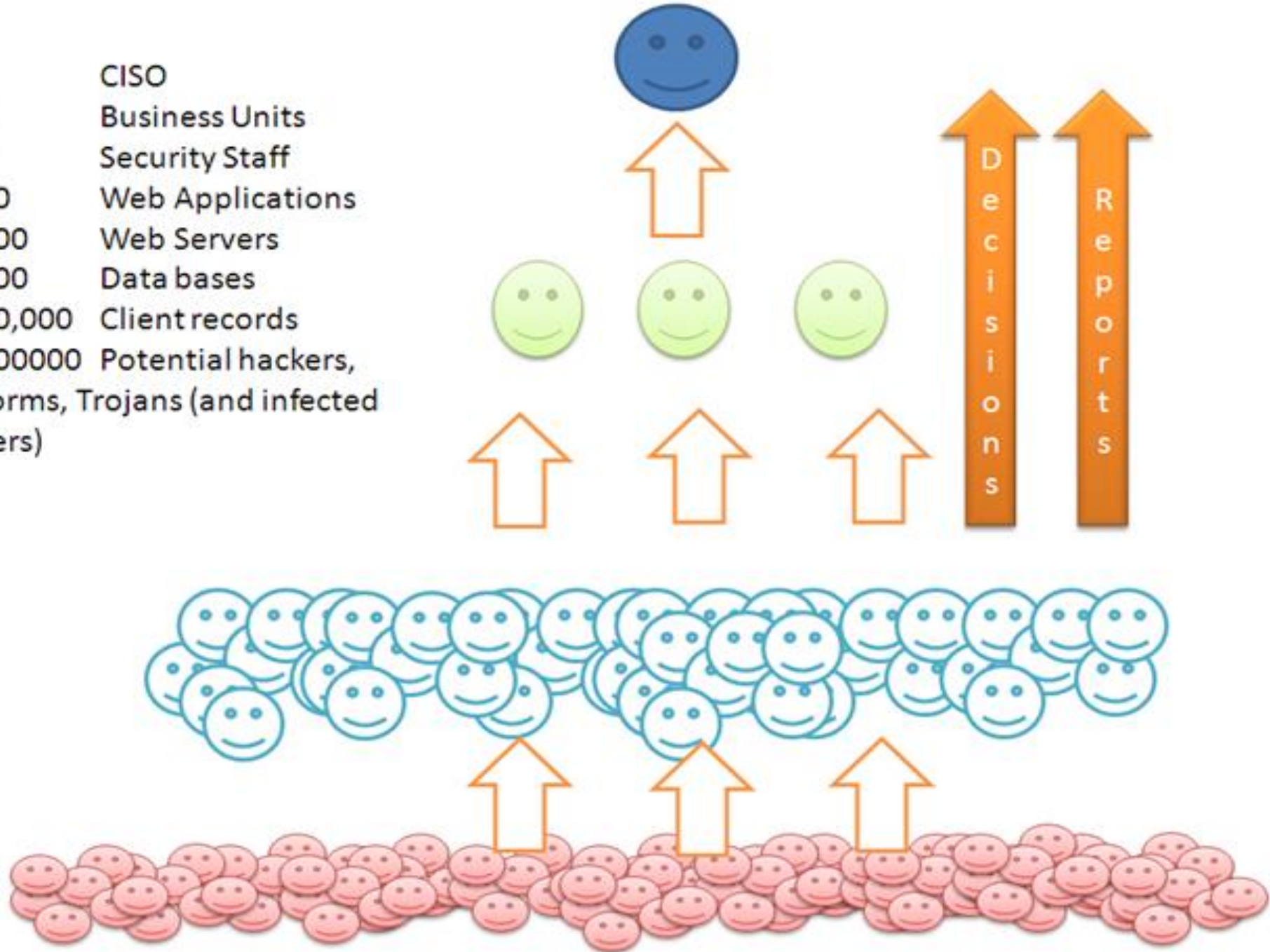
2.5.x – May 2017

<https://struts.apache.org/downloads.html>



Problem # 3
Bite off more than we can
chew

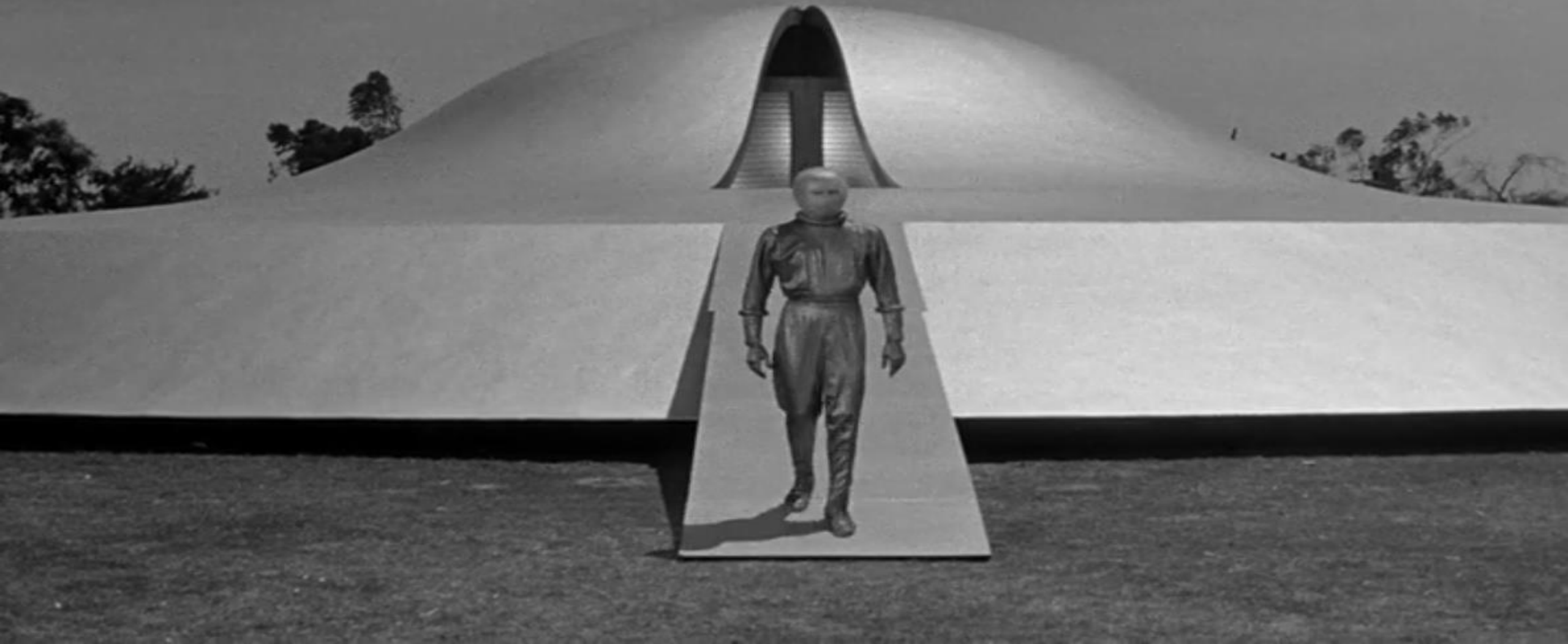
- 1 CISO
- 10 Business Units
- 30 Security Staff
- 200 Web Applications
- 1000 Web Servers
- 2000 Data bases
- 100,000 Client records
- 1000000 Potential hackers, Worms, Trojans (and infected users)



"We can't improve what we can't measure"

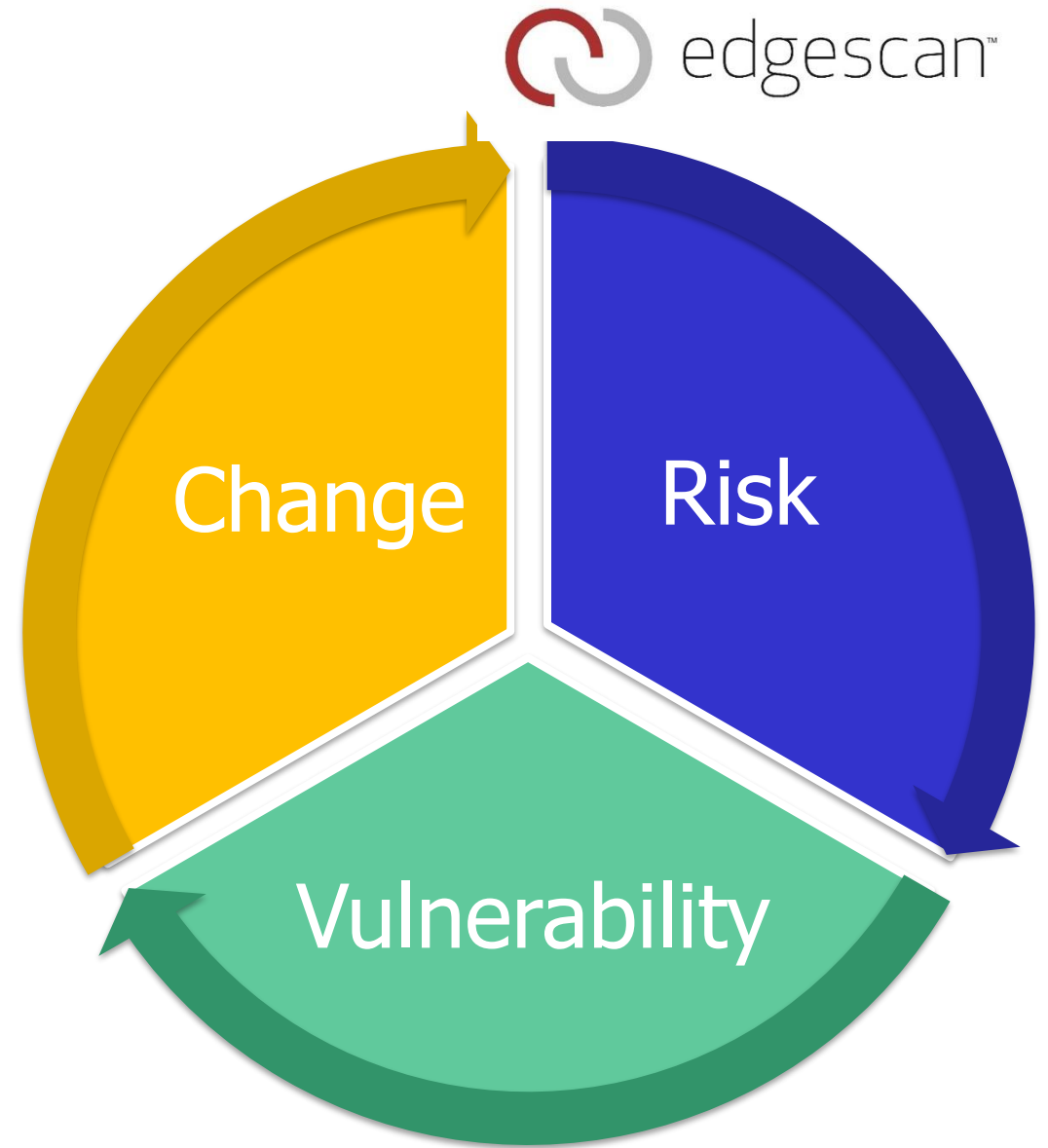
Problem # 4

Nothing ever stands still



Challenge

- Application Layer (Layer 7) is still more vulnerable.
- Applications change more.
- Change results in Risk (CI/CD/Agile)
- Risk (may) result in vulnerability & breach.



Keeping pace with change



- “Keeping pace” with development.
- Assisting secure **deployment**.
- Catching bugs **early** – Push Left.
- Help ensure “**change**” is secure.

TRADITIONAL APPROACH

Attacker Schedule



Defenders Schedule

CONTINUOUS APPROACH

Attacker Schedule



Defenders Schedule

Let's do a reality check



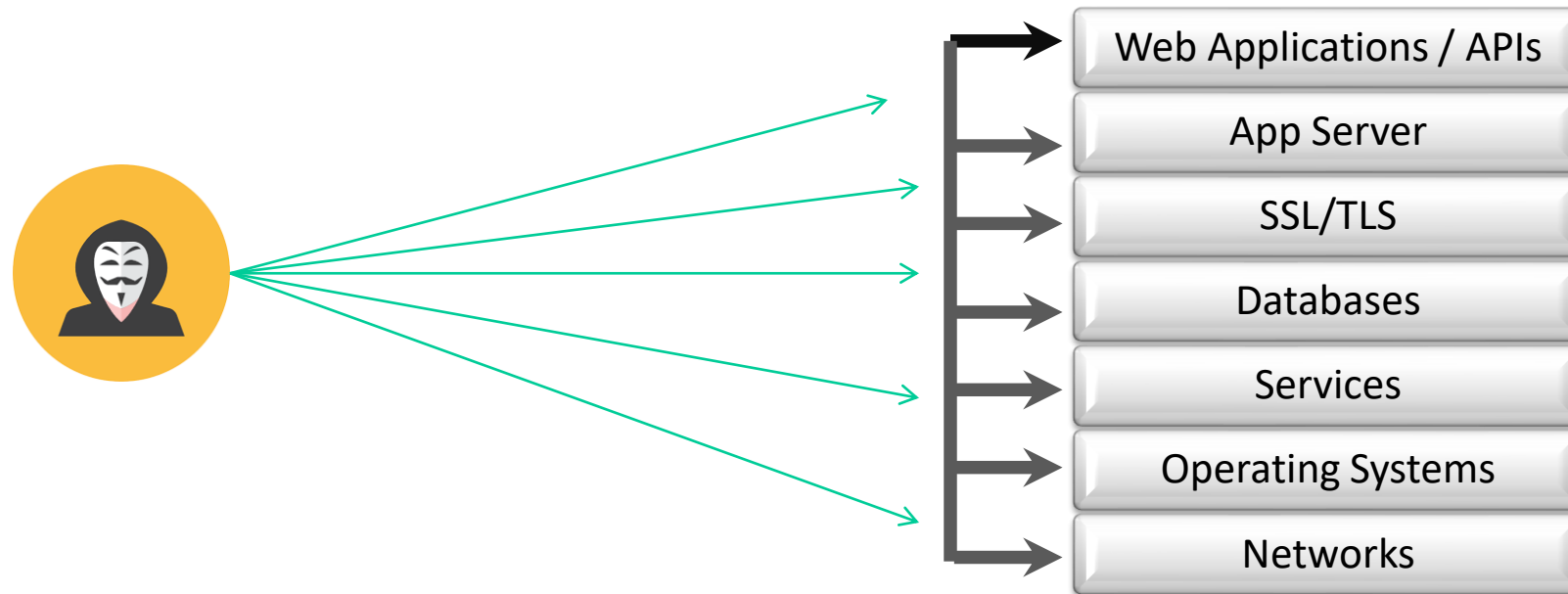
Where are these stats coming from?



- **edgescan™** is a sophisticated, enterprise-grade vulnerability assessment and management solution
- **edgescan™** helps from small & medium-sized to large enterprises identify and remediate known vulnerabilities
- **edgescan™** is a cloud based SaaS



#Fullstack



How we get the Statistical model



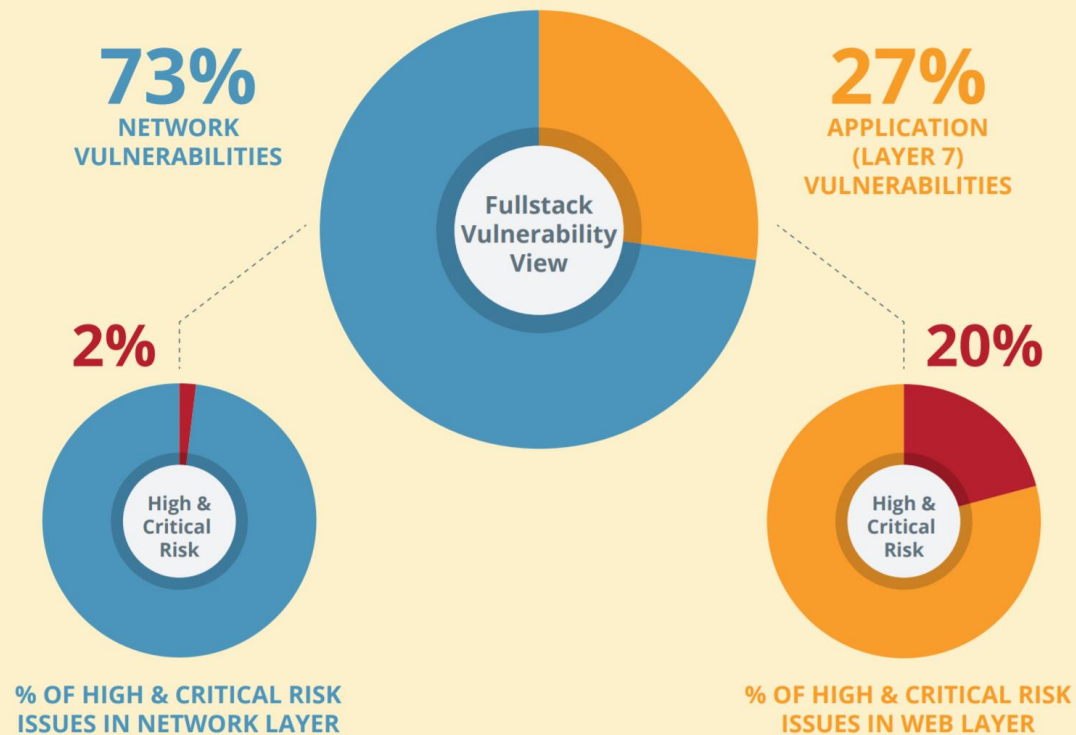
- 1000's of vulnerability assessments globally.
- #Fullstack view of security
- False positive free (99%) 😊
- Industries: Media, Energy, Government, Pharma, Finance, Software etc....



Risk Dispersion

FULLSTACK VULNERABILITY VIEW

In 2017 we discovered that on average, 27% of all vulnerabilities were associated with web applications and 73% were network vulnerabilities.



Web Application Layer (Layer 7)



Lots of high or critical risk issues!!

Easily exploitable

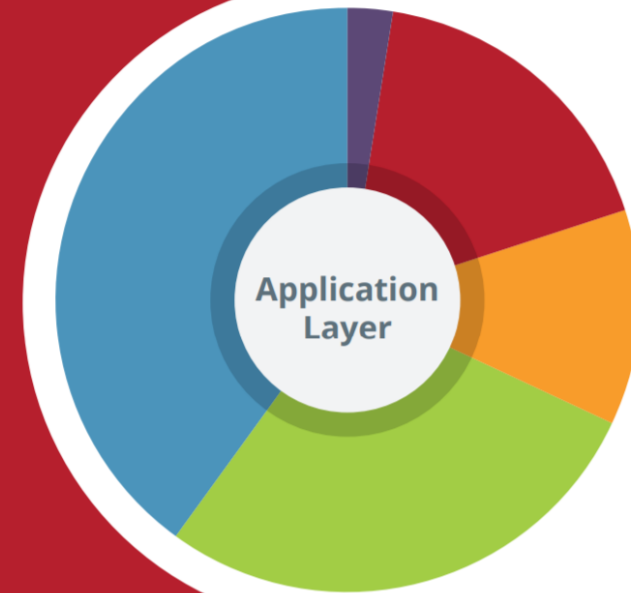
Very Damaging

Very Bad

APPLICATION LAYER RISK DENSITY

20% of all vulnerabilities discovered are High or Critical Risk

Every application is unique and developed uniquely which manifests in a high risk density.



2.7%
CRITICAL RISK

17.3%
HIGH RISK

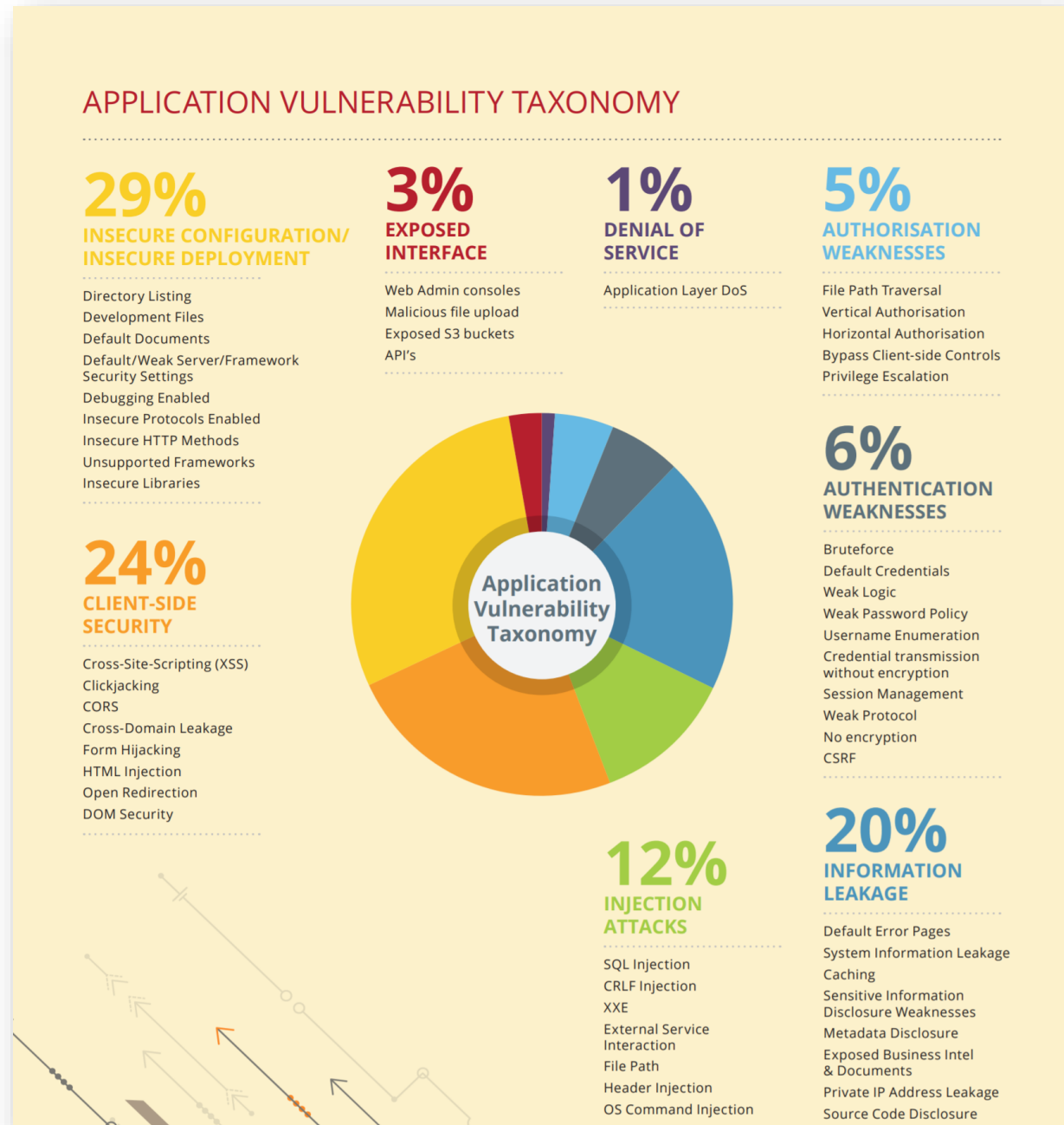
12%
MEDIUM RISK

28%
LOW RISK

40%
MINIMAL RISK

More Detail – App Layer

- System configuration and secure deployment is a big issue.
- Client-Side security: XSS, HTML Injection, Browser based issues are still very common.



Infrastructure Layer (Non Web app)

Lots of vulnerabilities!!

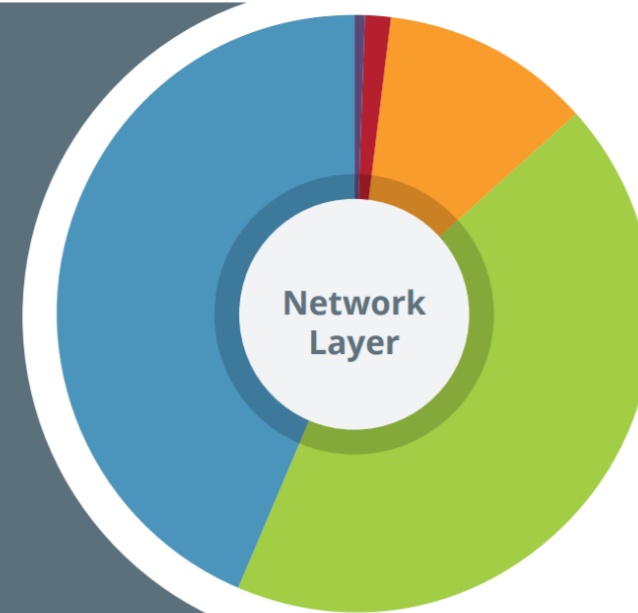
Not many high or Critical Risk.

More problems but less vulnerable

NETWORK LAYER RISK DENSITY

2% of all vulnerabilities discovered are High or Critical Risk

Hosting infrastructure and cloud is commoditised and appears to be easier to secure and maintain resulting in a lower percentage of high and critical risk density.



0.6%
CRITICAL RISK

1.5%
HIGH RISK

11.4%
MEDIUM RISK

43.5%
LOW RISK

43%
MINIMAL RISK

More Detail – Net Layer

- Large number of crypto-related issues: deprecated protocols, CVE's, poor implementation.
- Weak configuration / Mis-configuration

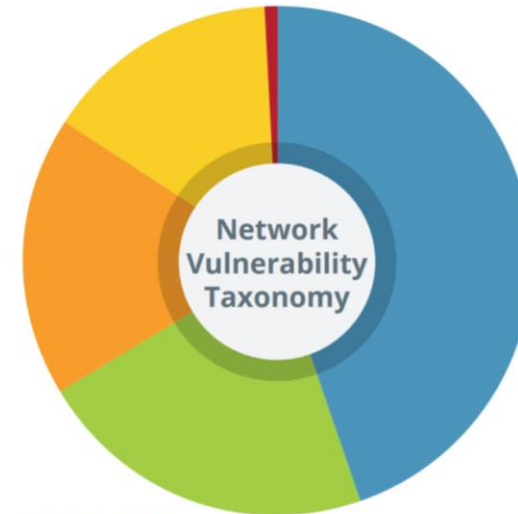
NETWORK VULNERABILITY TAXONOMY

<1%
EXPOSED SERVICES

Admin Consoles
RDP/Terminal Services
File Transfer
Sharepoint
RPC
Databases

15%
UNSUPPORTED

Microsoft IIS
Microsoft Outlook
MS 2003
OpenSSL
Samba
Deprecated SSL
Unsupported Unix
Unsupported Web Servers (IBM, Apache etc)



18%
PATCHING

Apache Vulnerabilities
Cisco Vulnerabilities
DNS Vulnerabilities
Firewall evasion
IKE Security Issues
IPMI Weaknesses
TCP/IP Stack Security
Microsoft Vulnerabilities
Open SSH Vulnerabilities
Open SSL Vulnerabilities
BSD Vulnerabilities
PHP Vulnerabilities
Wordpress Vulnerabilities

45%
CRYPTO

SSL/TLS/SSH – BREACH, SWEET, POODLE, DROWN, BEAST, CRIME
Short Keys Length
Weak Hashing
Weak Ciphers
RC4 Support

22%
CONFIGURATION

Default Credentials
FTP Exposure
HSTS Config
RDP Security
Weak SMB Config
Expired SSL/TLS certs
Misconfigured Certs
Terminal Services Security
Unencrypted/Telnet
Default Pages & Services
Lack of encryption

Known Vulnerabilities - Age



Patching and version maintenance is still a key part of maintaining a secure posture.

Known Vulnerabilities – Most Common



CVE-2004-2761

MD5 Message-Digest Algorithm is not collision resistant

CVSS: 5.0

CVE-2008-7220

Vulnerability in Prototype JavaScript framework (prototypejs)

CVSS: 7.5

CVE-2011-4969

Cross-site scripting (XSS) vulnerability in jQuery

CVSS: 4.3

CVE-2008-5161

Error handling vulnerability in the SSH protocol

CVSS: 2.6

Don't "Silo" / Segment Risk...

Hackers don't care where the vulnerability is!

Hackers
don't give a shit:



KIWICON III
28TH & 29TH NOVEMBER 2009

New Zealand's Hacker con - Wellington

- About your project's scope
- It's managed by a third party
- It's a legacy system
- It's "too critical to patch"
- About your outage windows
- About your budget
- You've always done it that way
- About your Go-Live Date
- It's only a pilot/proof of concept
- About Non-Disclosure Agreements
- It wasn't a requirement in the contract
- It's an internal system
- It's really hard to change
- It's due for replacement
- You're not sure how to fix it
- It's handled in the Cloud
- About your Risk Register entry
- The vendor doesn't support that configuration
- It's an interim solution
- It's [insert standard here] compliant
- It's encrypted on disk
- The cost benefit doesn't stack up
- "Nobody else could figure that out"
- You can't explain the risk to "The Business"
- You've got other priorities
- About your faith in the competence of your internal users
- You don't have a business justification
- You can't show Return on Investment
- You contracted out that risk



- Services
- Ports
- Patching (OS)
- Patching (Software Components)



- Vulnerabilities (Infrastructure) CVE
- Vulnerabilities (Unique) – Web Application
- Logical Vulnerabilities



Closing Comments – Be the Blockbuster!

1. Coverage
2. Stack
3. DevSecOps Integration
4. Min FP
5. Continuous / High Freq
6. Alerting
7. Metrics
8. API – squeeze that juice

—
—
**CONTINUOUS
VISIBILITY**



THANKS!

edgescan™ 2018 Vulnerability Stats Report:

2019
Coming
Soon!

Available now on:
edgescan.com

rahim@edgescan.com
@rahimjina



**NO
ANNOYING
REGISTRATION
NECESSARY!**