THE ART OF SECURING 100 PRODUCTS

Nir Valtman

@ValtmaNir

I work for NCR as the <HEAD> Application Security </HEAD>

1st time speaking publicly, except at blackhat DEFCON

BSIDES RSA CONFERENCE CSA cloud security alliance OWASP The Open Web Application Security Project Meetup

Mmmm…OH, AND

Neither of my previous startups succeeded!

But at least I invented few open source tools.

AntiDef

Secure TDD

Memory Scraper

SAPIA

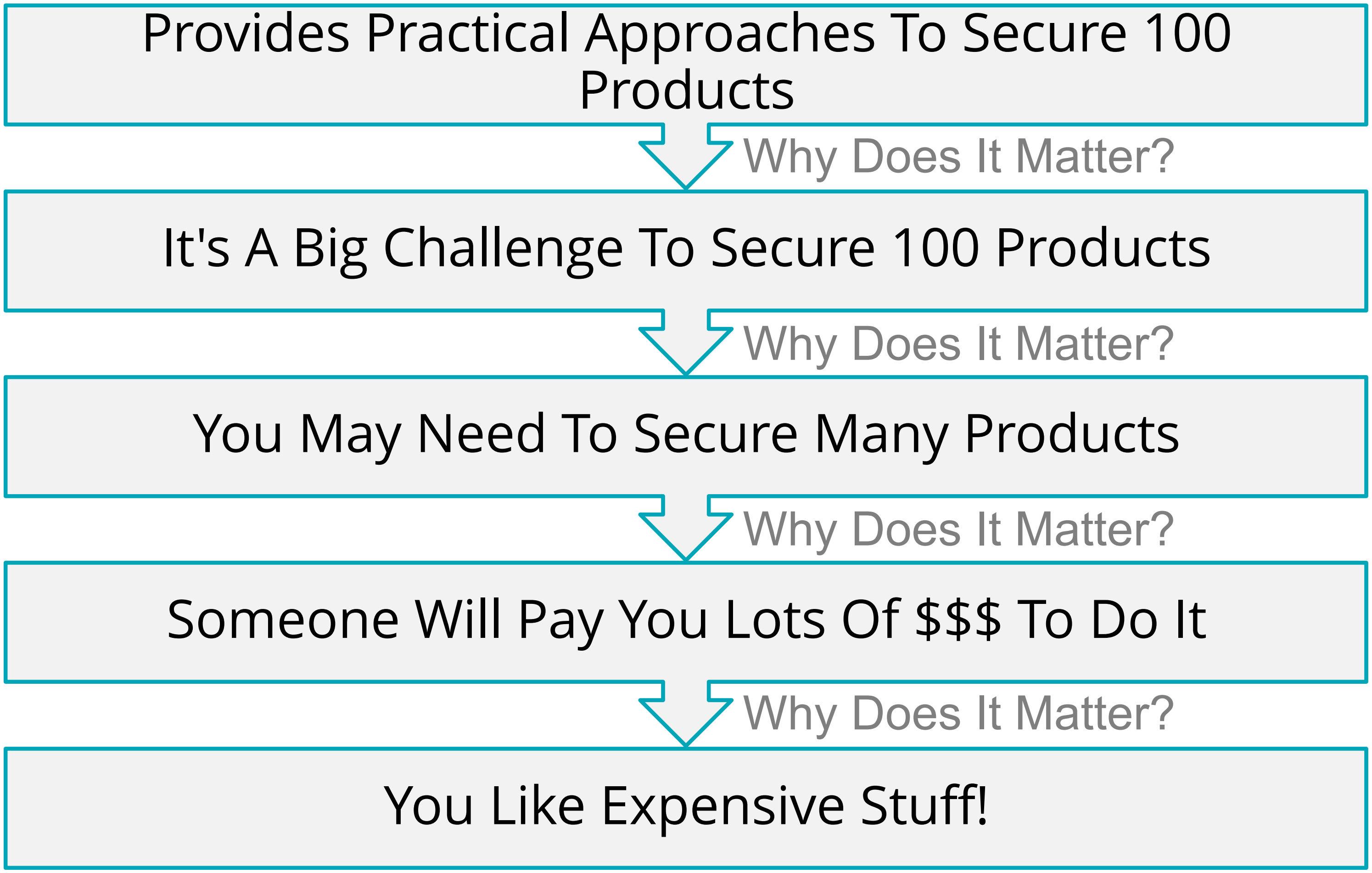
Cloudefigo

Lastly… I'm not a fan of the buzzword "Cyber"!

Cyber Cyber Cyber Arghhh!!!!

The trademark specified above is a trademark or a registered trademark of its respective company. This slide is intended for informational purposes only and does not represent any endorsement by this company.

Why Does This Talk Matter?



Provides Practical Approaches To Secure 100 Products

Why Does It Matter?

It's A Big Challenge To Secure 100 Products

Why Does It Matter?

You May Need To Secure Many Products

Why Does It Matter?

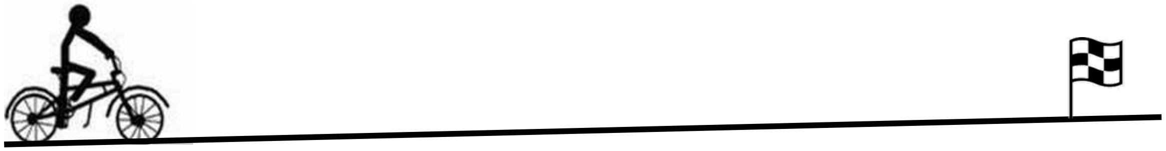
Someone Will Pay You Lots Of $$$ To Do It

Why Does It Matter?

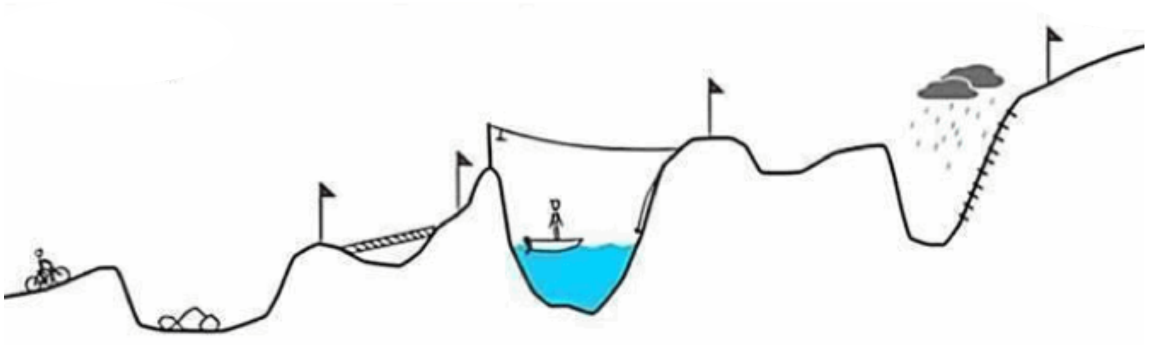
You Like Expensive Stuff!

Agenda

Plan



Reality



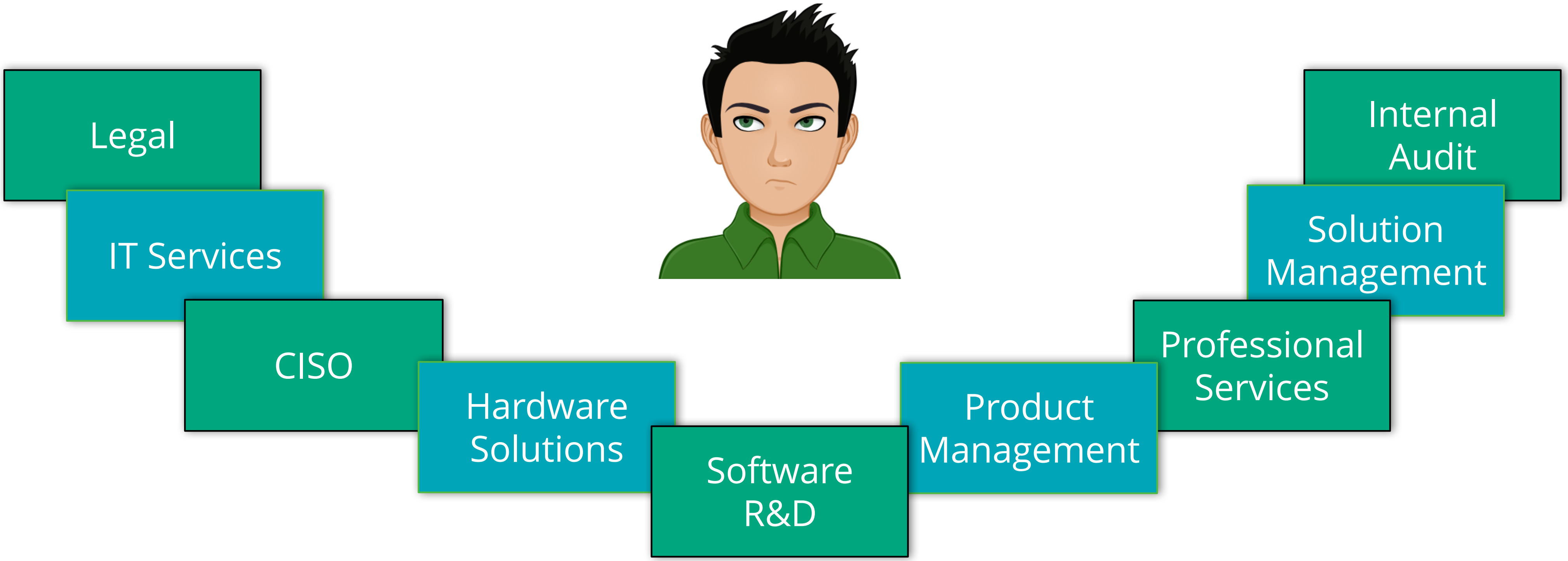
Meet The Application Security Lead!

- Accountable for Product Security
  - Cloud-based, self-hosted or installed on customers’ premise
  - Part of the products are regulated
- Needs to keep the company out of the news
- Got executive leadership to support him

* Avatar generated on avatarmaker.com

The Daily Challenges

Need to secure a *single* high-risk product. Who’s involved?



Legal

IT Services

CISO

Hardware Solutions

Software R&D

Product Management

Professional Services

Solution Management

Internal Audit

Mapping The Business Owners

Product #1

- ✓ Software R&D
- ✓ CISO
- ✓ Legal
- ✓ Product Management
- ✓ Internal Audit

Product #2

- ✓ Software R&D
- ✓ CISO
- ✓ Legal
- ✓ Product Management
- ✓ Solution Management
- ✓ Hardware Solutions

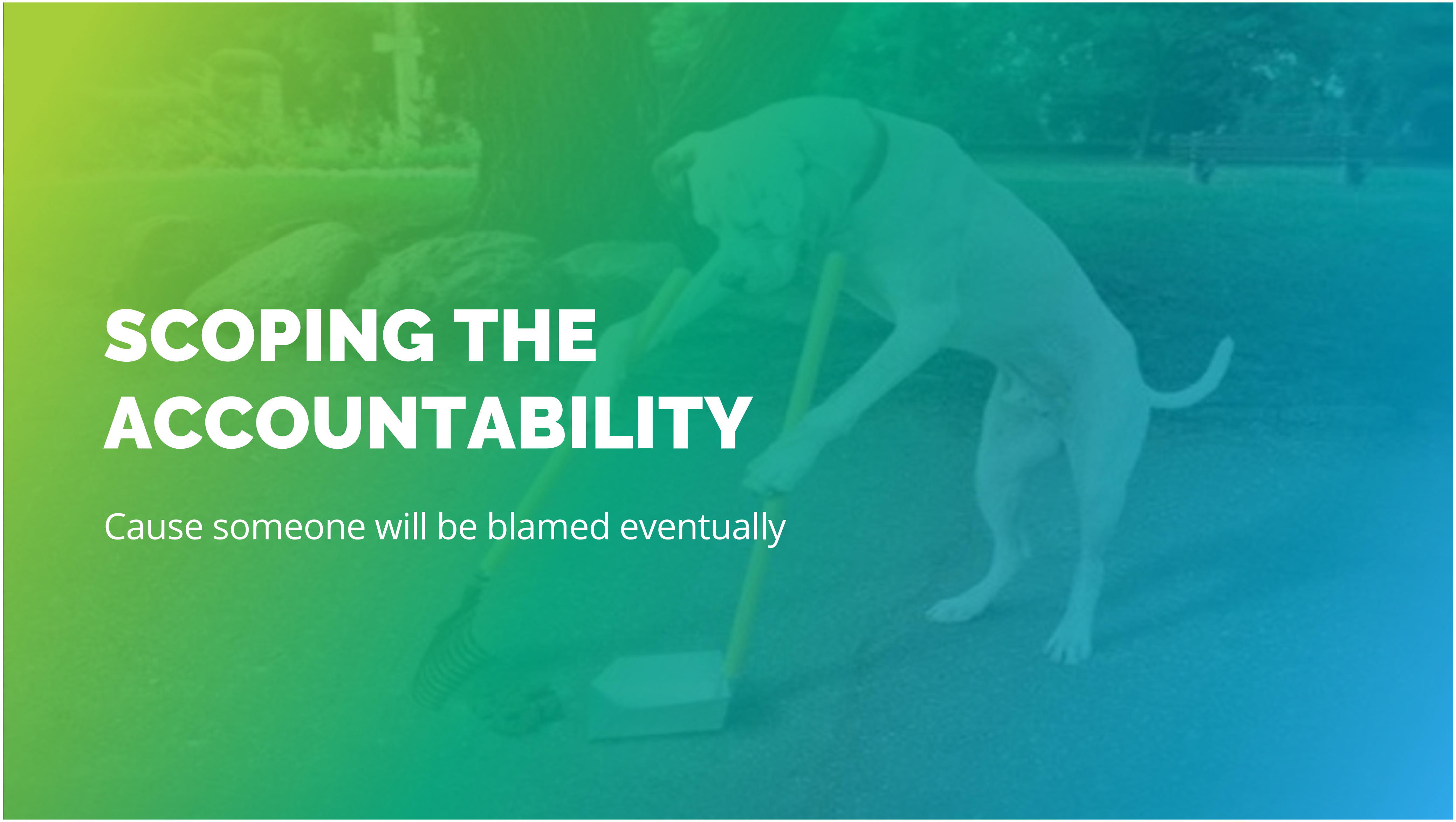
Product #100

- ✓ Software R&D
- ✓ IT
- ✓ CISO
- ✓ Legal
- ✓ Product Management
- ✓ Solution Management
- ✓ Professional Services

Will I Finish This Mapping Soon?

Start Date

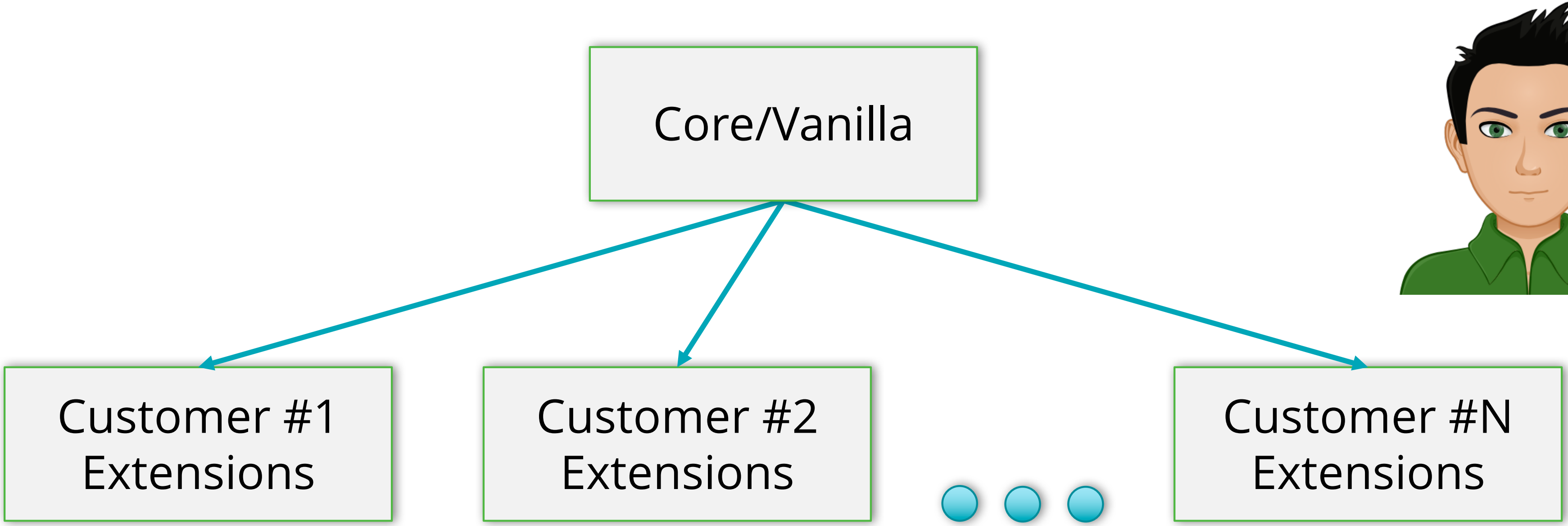
Done!



SCOPING THE ACCOUNTABILITY

Cause someone will be blamed eventually

Core vs. Extensions

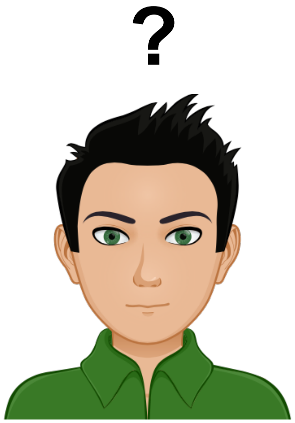


Core/Vanilla

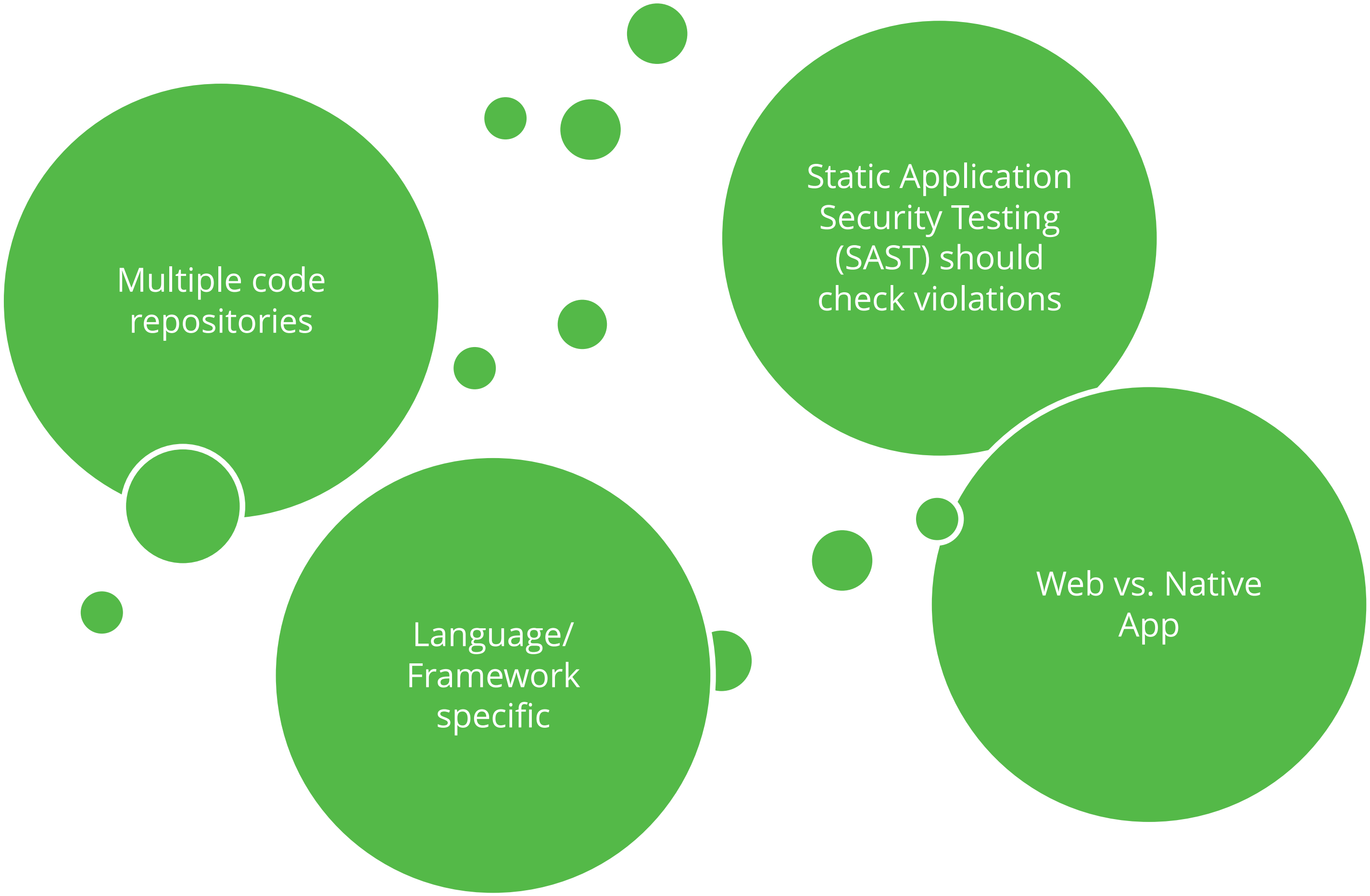
Customer #1 Extensions

Customer #2 Extensions

Customer #N Extensions



In theory, building a central security library is a best practice. In practice, theory sucks!



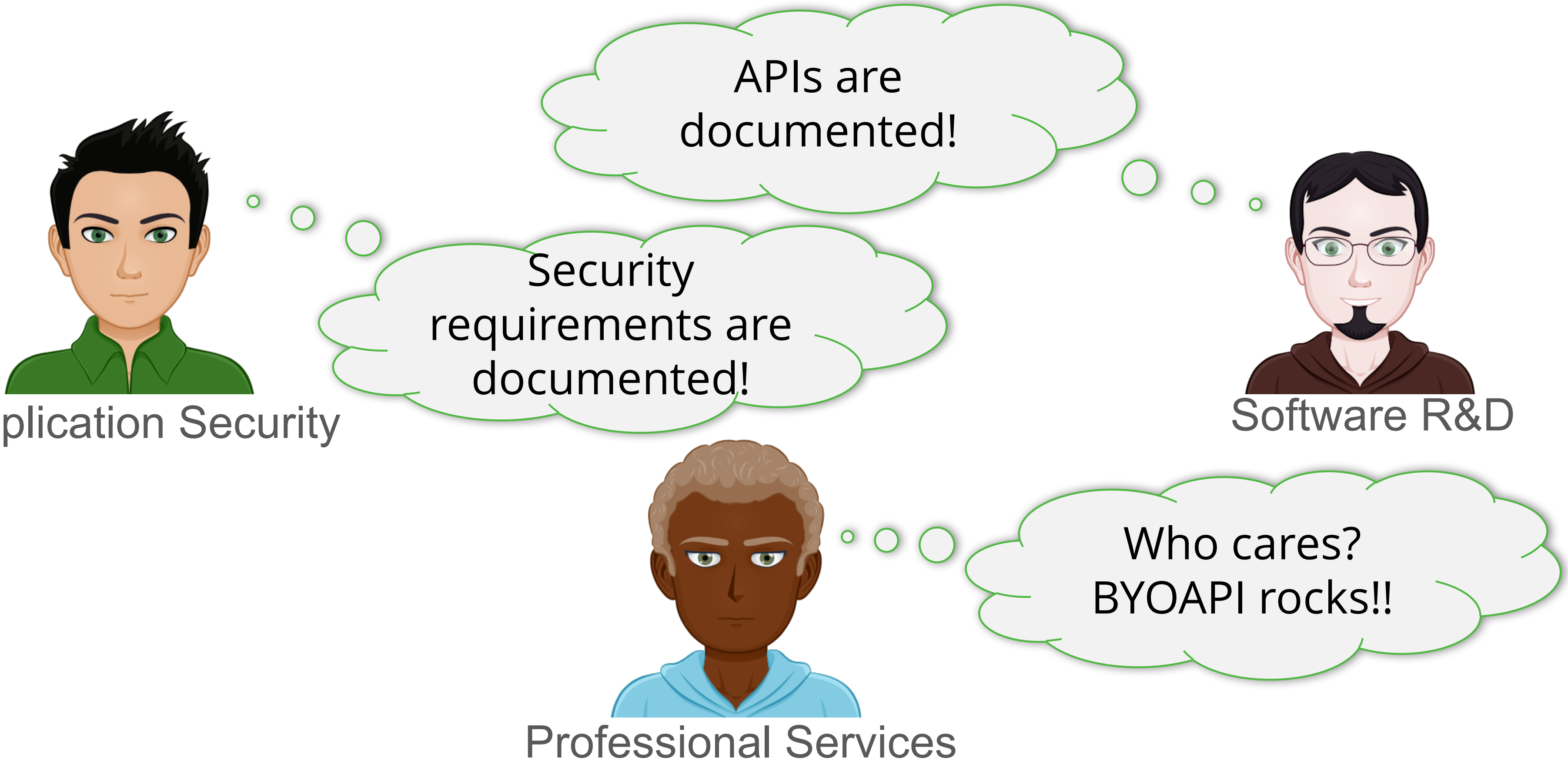
Multiple code repositories

Static Application Security Testing (SAST) should check violations

Language/ Framework specific

Web vs. Native App

Difficult To Control All Engineering Parties



Application Security

Security requirements are documented!

Software R&D

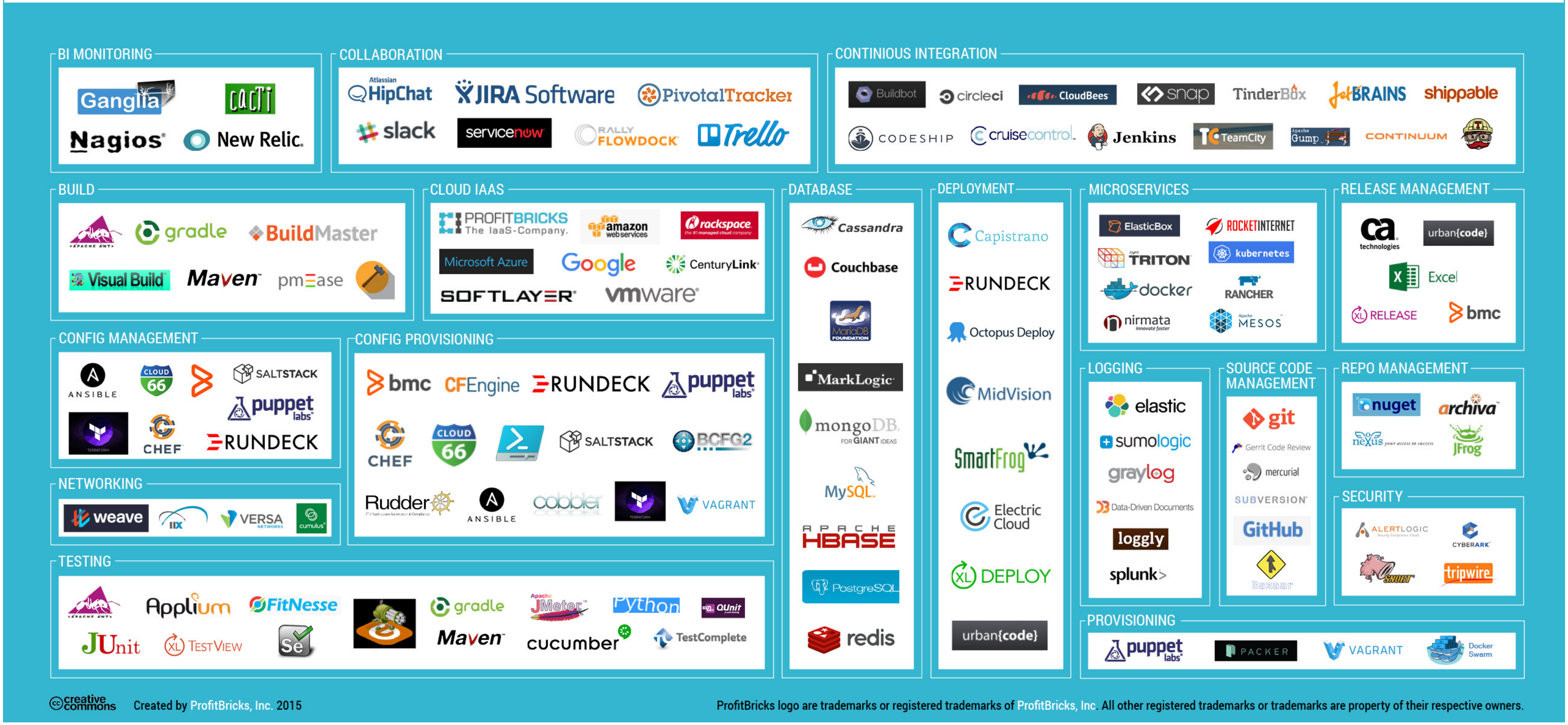
APIs are documented!

Professional Services

Who cares? BYOAPI rocks!!

RESOURCE DIVERSITY & LIMITATIONS

PROFITBRICKS PRESENTS:
LANDSCAPE VIEW OF INFRASTRUCTURE AUTOMATION COMPANIES
LATE 2015 (V 1.0)



BI MONITORING
Ganglia, Cacti, Nagios, New Relic.

COLLABORATION
Atlassian HipChat, JIRA Software, PivotalTracker, slack, servicenow, Rally Flowdock, Trello

CONTINIOUS INTEGRATION
Buildbot, circleci, CloudBees, snap, TinderBox, JetBRAINS, shippable, CODESHIP, cruisecontrol, Jenkins, TeamCity, Apache Gump, CONTINUUM

BUILD
Apache Ant, gradle, BuildMaster, Visual Build, Maven, pmEase

CLOUD IAAS
PROFITBRICKS The IaaS-Company., amazon web services, rackspace, Microsoft Azure, Google, CenturyLink, SOFTLAYER, vmware

DATABASE
Cassandra, Couchbase, MariaDB Foundation, MarkLogic, mongoDB for GIANT IDEAS, MySQL, Apache HBASE, PostgreSQL, redis

DEPLOYMENT
Capistrano, RUNDECK, Octopus Deploy, MidVision, SmartFrog, Electric Cloud, XL DEPLOY, urban{code}

MICROSERVICES
ElasticBox, ROCKETINTERNET, TRITON, kubernetes, docker, RANCHER, nirmata, Apache MESOS

RELEASE MANAGEMENT
ca technologies, urban{code}, Excel, XL RELEASE, bmc

CONFIG MANAGEMENT
ANSIBLE, CLOUD 66, SALTSTACK, puppet labs, TERRAFORM, CHEF, RUNDECK

CONFIG PROVISIONING
bmc, CFEngine, RUNDECK, puppet labs, CHEF, CLOUD 66, SALTSTACK, BCFG2, Rudder, ANSIBLE, cobbler, TERRAFORM, VAGRANT

NETWORKING
weave, IIX, VERSA NETWORKS, cumulus

LOGGING
elastic, sumologic, graylog, Data-Driven Documents, loggly, splunk>

SOURCE CODE MANAGEMENT
git, Gerrit Code Review, mercurial, SUBVERSION, GitHub, Bazaar

REPO MANAGEMENT
nuget, archiva, nexus your access to success, JFrog

SECURITY
ALERTLOGIC, CYBERARK, SNORT, tripwire

TESTING
Apache Ant, Appium, FitNesse, gradle, Apache JMeter, Python, QUnit, JUnit, XL TestView, Se, Maven, cucumber, TestComplete

PROVISIONING
puppet labs, PACKER, VAGRANT, Docker Swarm

Created by ProfitBricks, Inc. 2015

The trademarks specified above are trademarks or registered trademarks of their respective owners. This slide is intended for informational purposes only and does not represent any endorsement

Diverse Application Security Tools

Static Application Security Testing (SAST)	CHECKMARX IBM Security Hewlett Packard Enterprise SYNOPSYS® CONTRAST SECURITY VERACODE WhiteHat SECURITY.
Dynamic Application Security Testing (DAST)	Hewlett Packard Enterprise VERACODE IBM Security SYNOPSYS® RAPID7 netsparker BURPSUITE PROFESSIONAL acunetix
Interactive Application Security Testing (IAST)	CONTRAST SECURITY SYNOPSYS® WhiteHat SECURITY. IBM Security VERACODE
Software Composition Analysis	WhiteSource BLACKDUCK VERACODE SYNOPSYS®

& More...

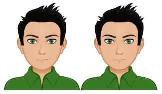
Mobile AST

Container Security

Code Obfuscation

The trademarks specified above are trademarks or registered trademarks of their respective owners. This slide is intended for informational purposes only and does not represent any endorsement.

Labor Limitations



1%-2%

of engineering org size

APPLICATION SECURITY MATURITY PROGRAM

Maturity is knowing when and where to be immature

Governance – Easy To Say, Difficult To Control

Develop an S-SDLC	Enforce the S-SDLC
Provide Technology-Specific Training	Map, Track & Drive Towards Completion Of Trainings
Define Risk Management & Risk Acceptance Process	Get Executives To Sign On A Security Risk

Construction – Relatively Difficult

Threat Assessment

Documenting risks in agile development lifecycle consumes much resources

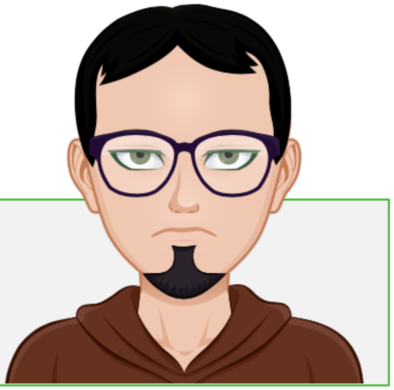
Security Requirements

Should app security be involved in ALL requirements sessions?

Security Architecture

Providing best practices for various product types

Verification – Roadblocks Ahead!



Design Review

- Get a design diagram from engineering teams… *lots* of teams!!!
- Working with *many* smart engineering people – they know everything!

Code Review

- Utilizing automation is great if *ALL* bug tracking, code repo, and build systems are centralized
- Scaling automation for 100 products is nearly impossible (technology & labor wise)
- Building a central security library is a waste of time if technologies are vary!

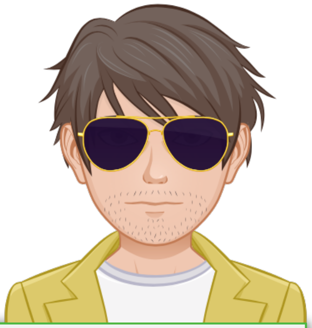
Security Testing

- Automation = [sophisticated] vulnerability scanning. Manual work = penetration test!
- $$$

Deployment – Only Sounds Easy

”Shadow”-operated IRT

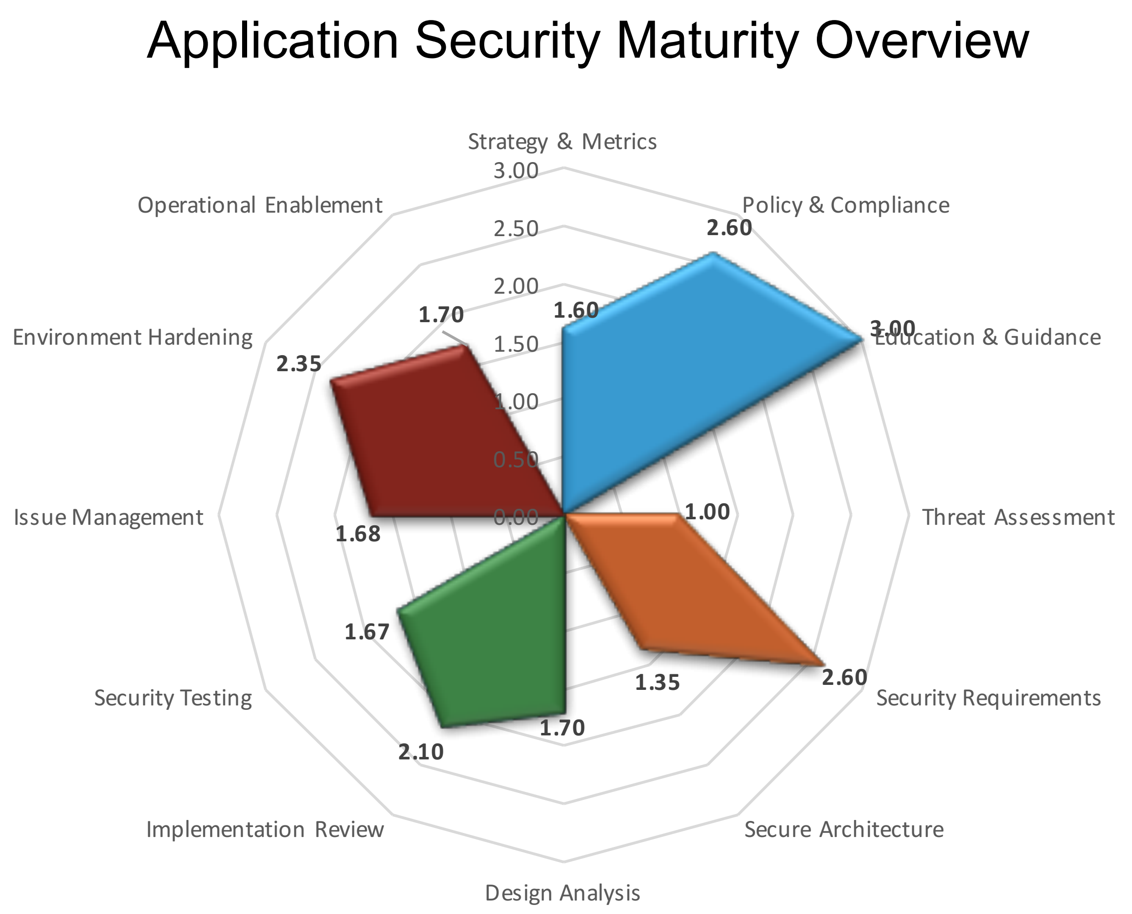
Work w/ every engineering team to QA hardening



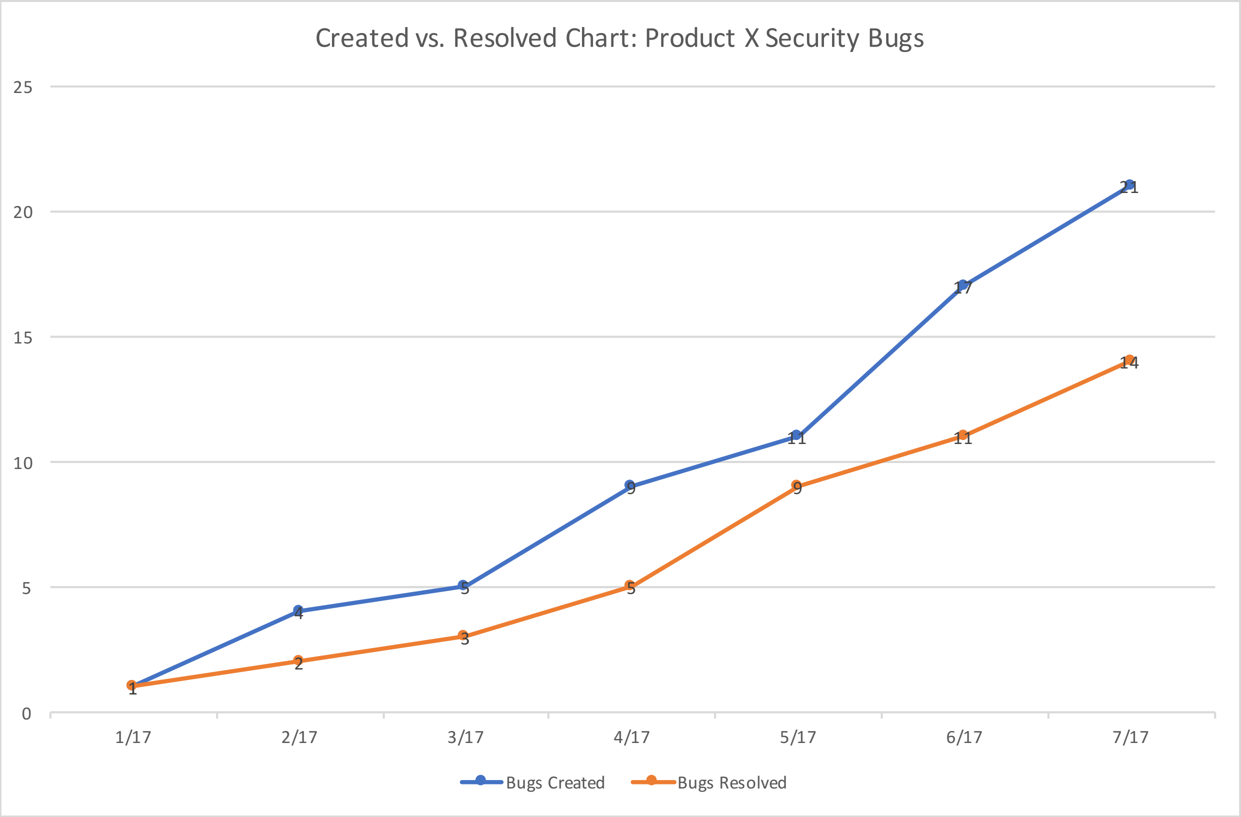
Corporate IRT

If I define & It doesn’t work, ⇦ they’re responsible

Perspective On 1 Of 100 Products



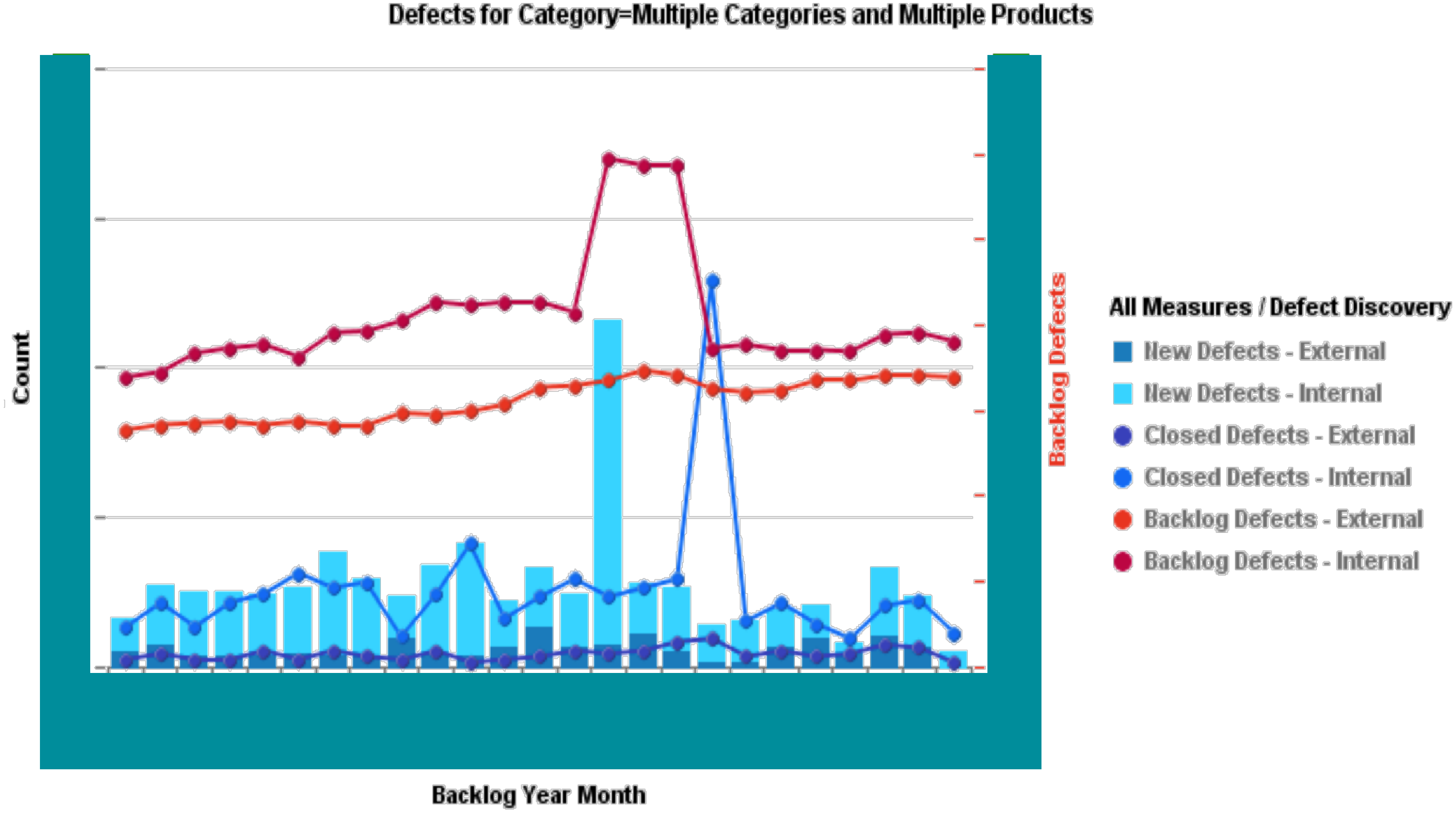
Perspective On 1 Of 100 Products



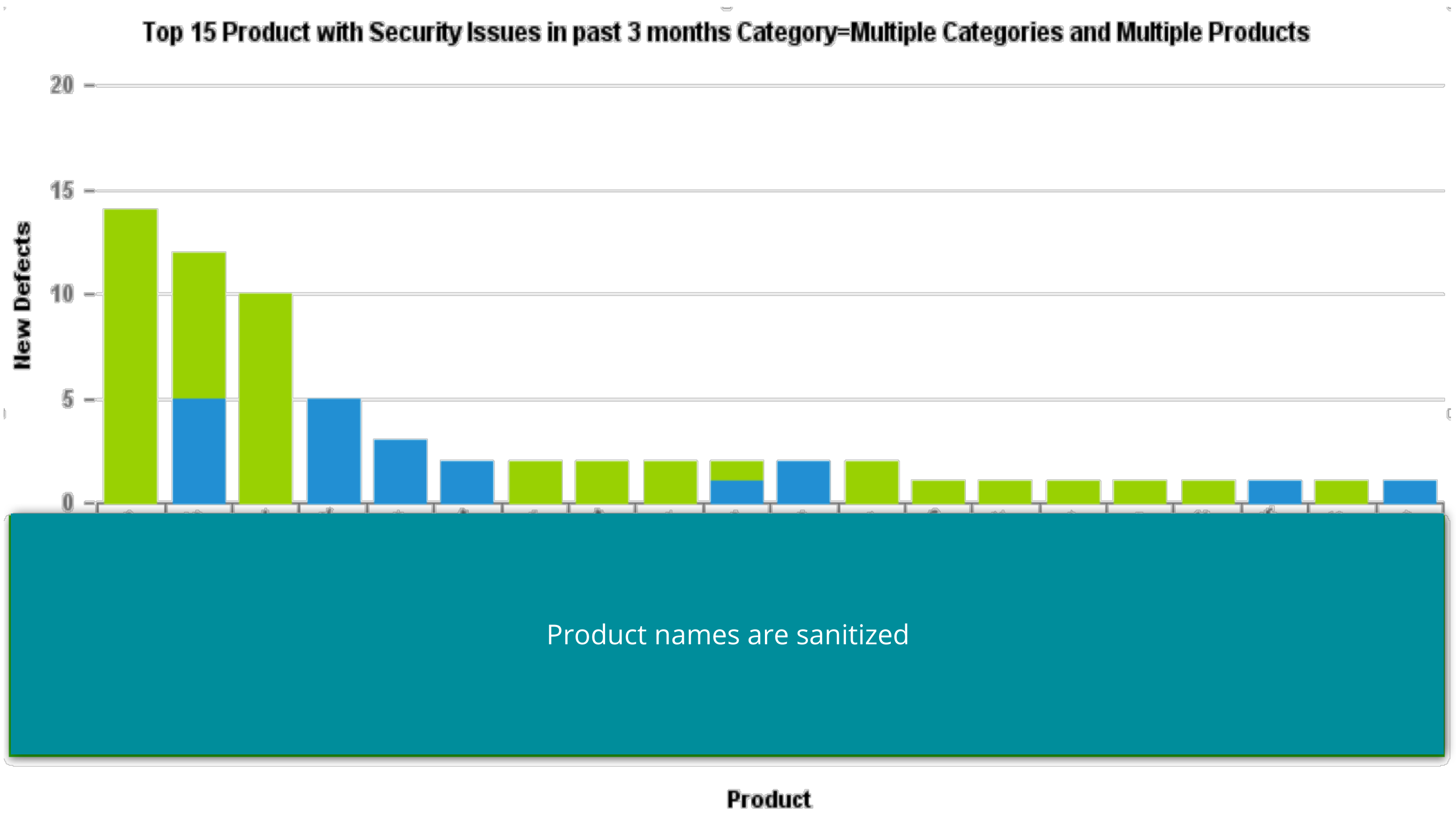
Additional Considerations

- Number of items per development lifecycle stage
  - E.g. pending QA, not started, in dev, etc.
- Average time to mitigate a vulnerability
- Prioritized list of outstanding Epics/US/Bugs

Perspective On 100 Products



Perspective On 100 Products

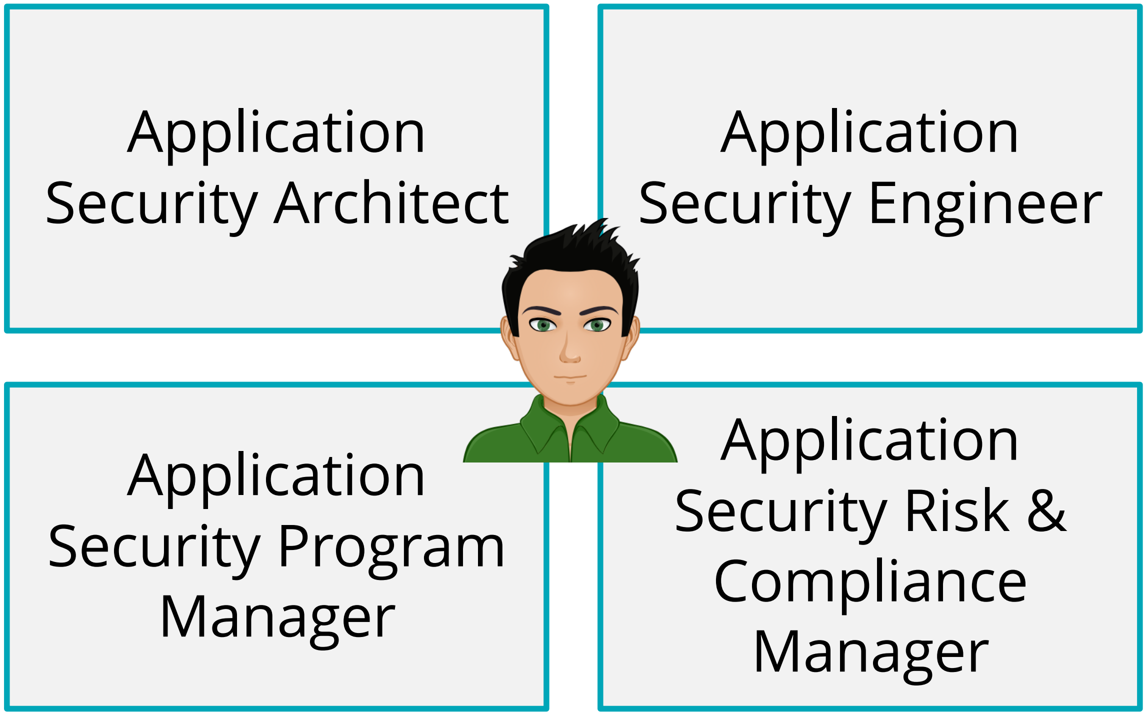


Product names are sanitized

DON’T REINVENT THE WHEEL, JUST REALIGN IT

(Anthony J. D’Angelo)

NCR's App Sec Team's Specialties



Application Security Architect

Application Security Engineer

Application Security Program Manager

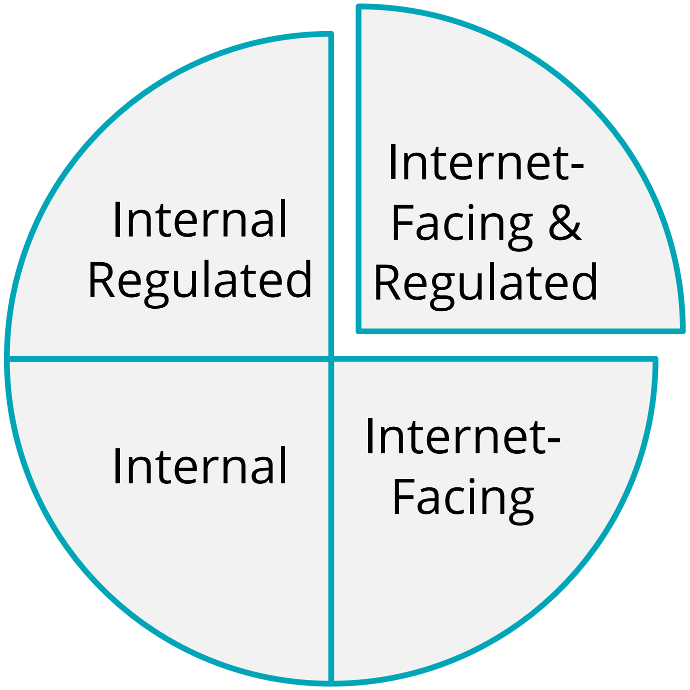
Application Security Risk & Compliance Manager

NCR’s App Sec Team’s Specialties Mapping

OpenSAMM		Speciality			
Domain	Activity	Security Architecture	Program Management	Risk Management & Compliance	Application Security Engineering
Governance	Strategy & Metrics		V	V	
	Policy & Compliance			V	
	Education & Guidance	V	V	V	V
Construction	Threat Assessment	V			
	Security Requirements	V	V	V	
	Secure Architecture	V			
Verification	Design Review	V			
	Code Review	V			V
	Security Testing				V
Deployment	Vulnerability Mgmt		V	V	
	Environment Hardening		V		
	Operational Enablement		V		

Prioritizing Security

Product Type



Internal Regulated

Internet-Facing & Regulated

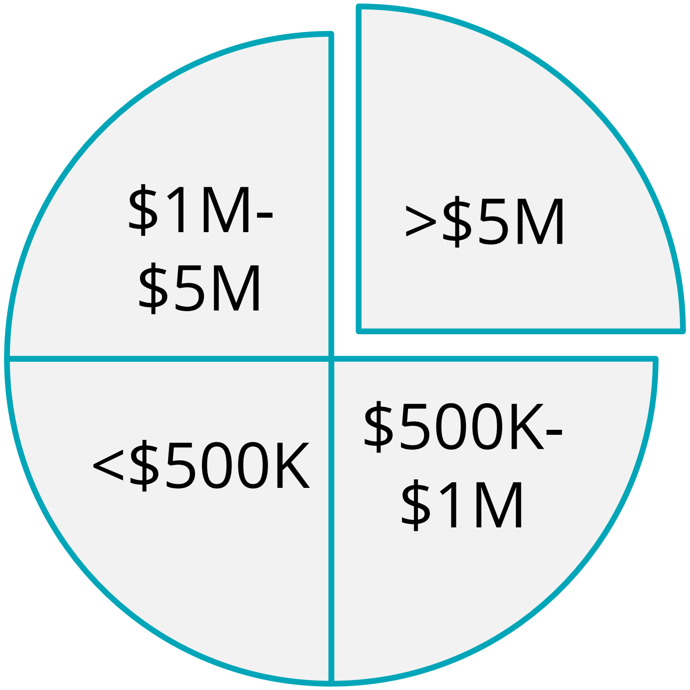
Internal

Internet-Facing

Strategy

Investments & Commitments

Financial Impact



$1M-$5M

>$5M

<$500K

$500K-$1M

Budgeting Labor Correctly – The Formula

Product Type	% Of R&D
Internet-Facing & Regulated	2%
Internet-Facing	1%
Internal & Regulated	1%
Internal	0.3%

×

R&D Labor Count

×

Product Type	% Of AppSec
Program Manager	20%
Risk & Compliance	10%
Architecture	23%
Engineering	47%

Example

An *Internet-facing & regulated* product suite that is developed by an org size of *1000 employees* needs:

2% X 1000 = *20 App Sec Team Members*, consisting of 4 PM, 2 R&C, 4.6 Architects and 9.4 Engineers

A Lesson Learned

Even with an aggressive strategy, hiring app sec people is a REAL bottleneck!

A Satellite Program

Give a poor man a fish and you feed him for a day. Teach him to fish and you give him an occupation that will feed him for a lifetime.”

(Chinese proverb.)

A Satellite Program Example

	Yellow Belt	Green Belt	Brown Belt	Black Belt
Online Training	Foundation app sec classes	Advanced classes		
Instructor-led or conferences participation		Various advanced topics		
Special Interest		PII, GDPR, PCI, FFIEC		
Static/Dynamic/Interactive Security Analysis	On-boarding		Tool/Process improvement	
Advanced			Threat modeling	Standards review, reusable IP

Measuring Effectiveness!

Ongoing

- Escalations asking for security resources by the engineering teams are good!
- Status reports must be balanced
  - Neither too Green nor Red

	🟥	
	🟧	
	🟧	
	🟧	
	🟩	

Measuring Effectiveness!

Year Over Year

- Overall Application Security Maturity rank increases
- Decreased number of security vulnerability reporting per X (you to define) lines of code
  - Engineers will always make mistakes
  - Use 3rd parties to assess it

Scaling Out Team's Capabilities

Security Questionnaire For Engaging An App Sec Architect

10 Yes/No Questions

Scaling Out Team’s Capabilities

Security Questionnaire For Engaging An App Sec Architect

10 Yes/No Questions

Is Data Classified?

Handle Sensitive Data Related To PII/PCI?

Do You Follow The S-SDLC?

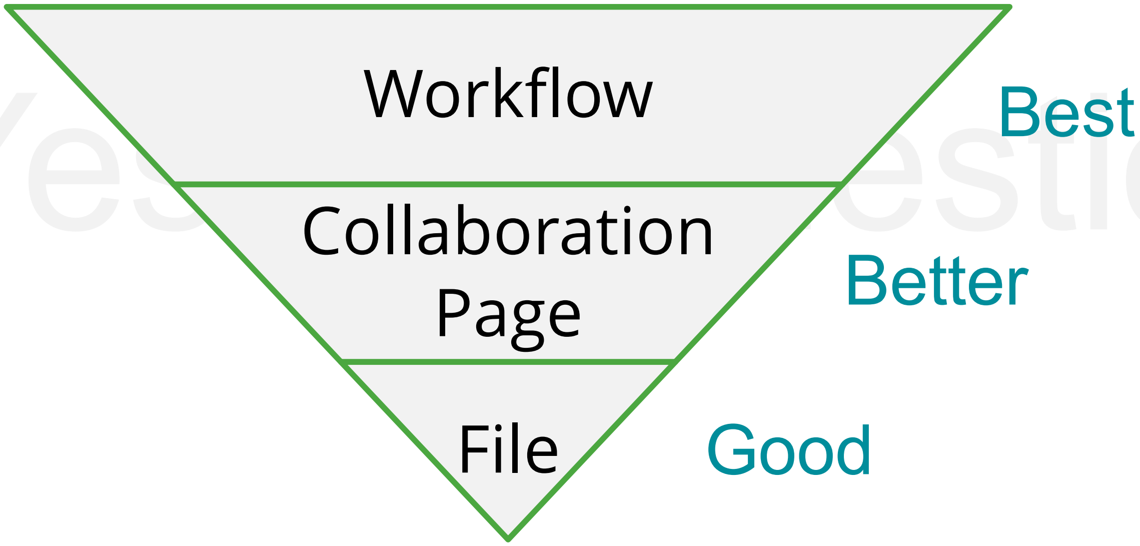
Security Automation Integrated Into Pipeline?

Data Encryption?

Consumer-Facing Mobile App?

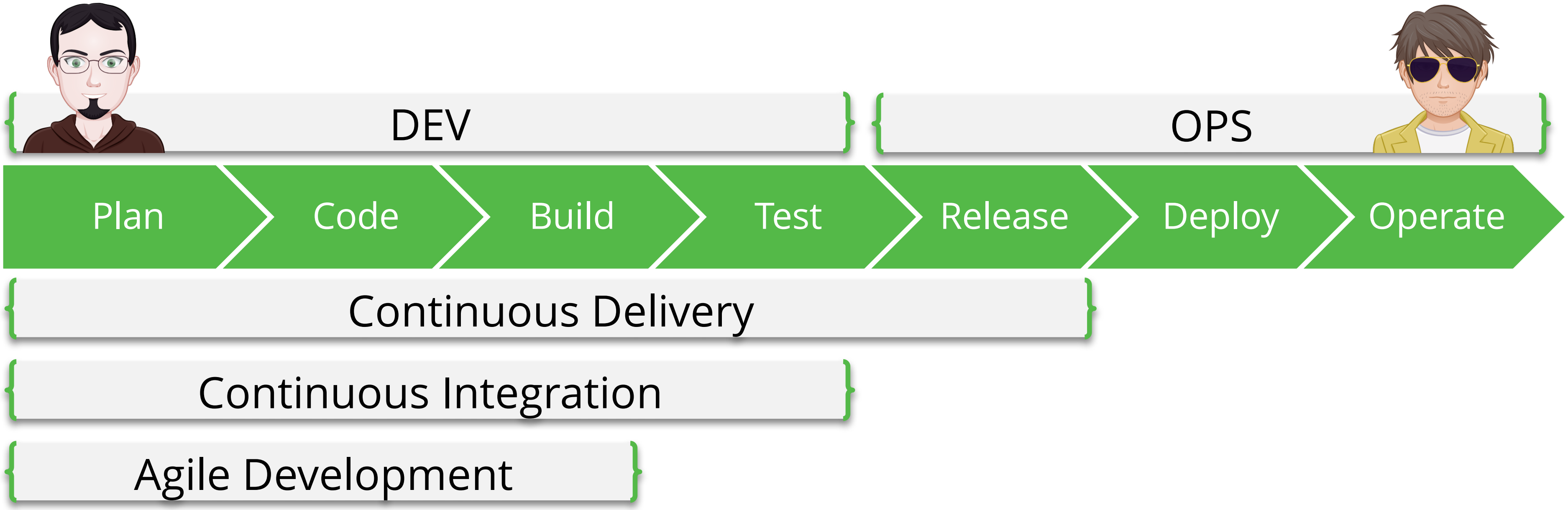
Scaling Out Team’s Capabilities

Security Questionnaire For Engaging An App Sec Architect



Scaling Up Security

Application Security Must Fit Into Any Pipeline



DEV

OPS

Plan

Code

Build

Test

Release

Deploy

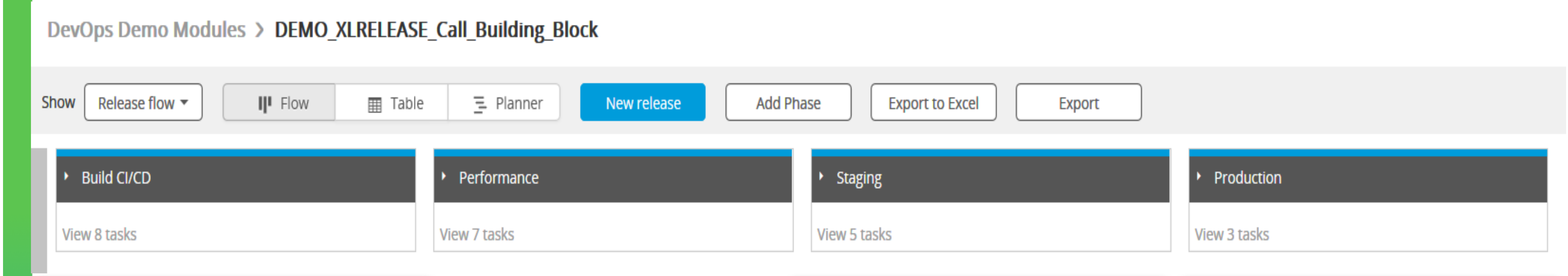
Operate

Continuous Delivery

Continuous Integration

Agile Development

Scaling Up Security Using Release Automation



Static App Sec Testing

Interactive App Sec Testing (IAST)

Binary Signing

Code Obfuscation

Dynamic App Sec Testing (DAST)

Vulnerability Scanning

Runtime App Self Protection (RASP)

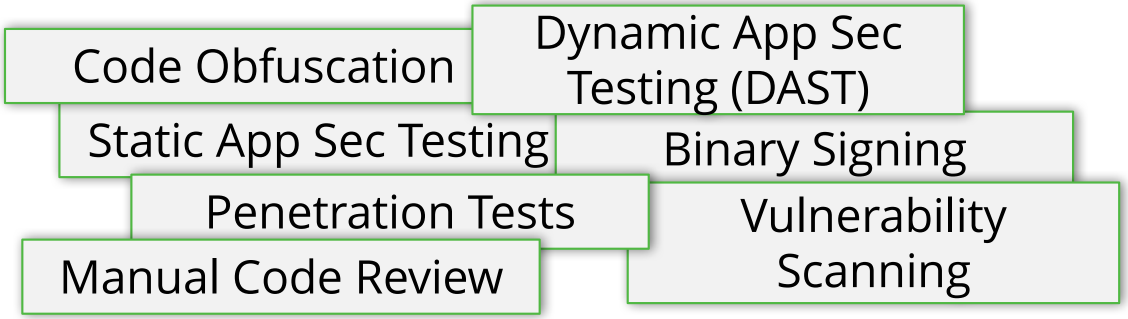
Dynamic App Sec Testing (DAST)

Vulnerability Scanning

Runtime App Self Protection (RASP)

Scaling Up Security When Lacking Automation

Identify Quick Wins



Code Obfuscation

Dynamic App Sec Testing (DAST)

Static App Sec Testing

Binary Signing

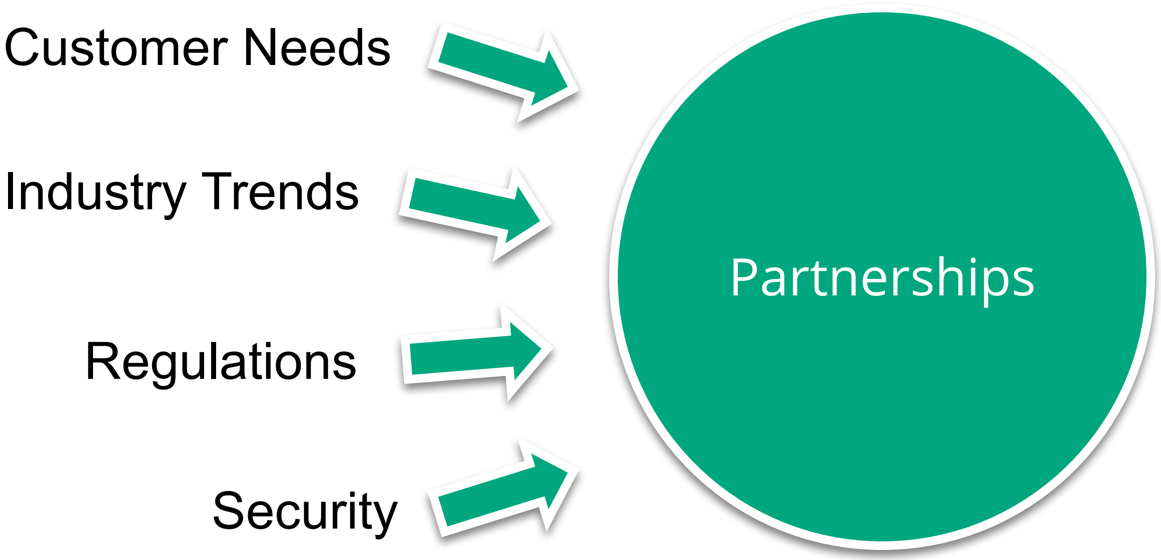
Penetration Tests

Vulnerability Scanning

Manual Code Review

Even A Long-Term Plan Is A Viable Plan

Finding The Partnerships – Use Cases



Customer Needs

Industry Trends

Regulations

Security

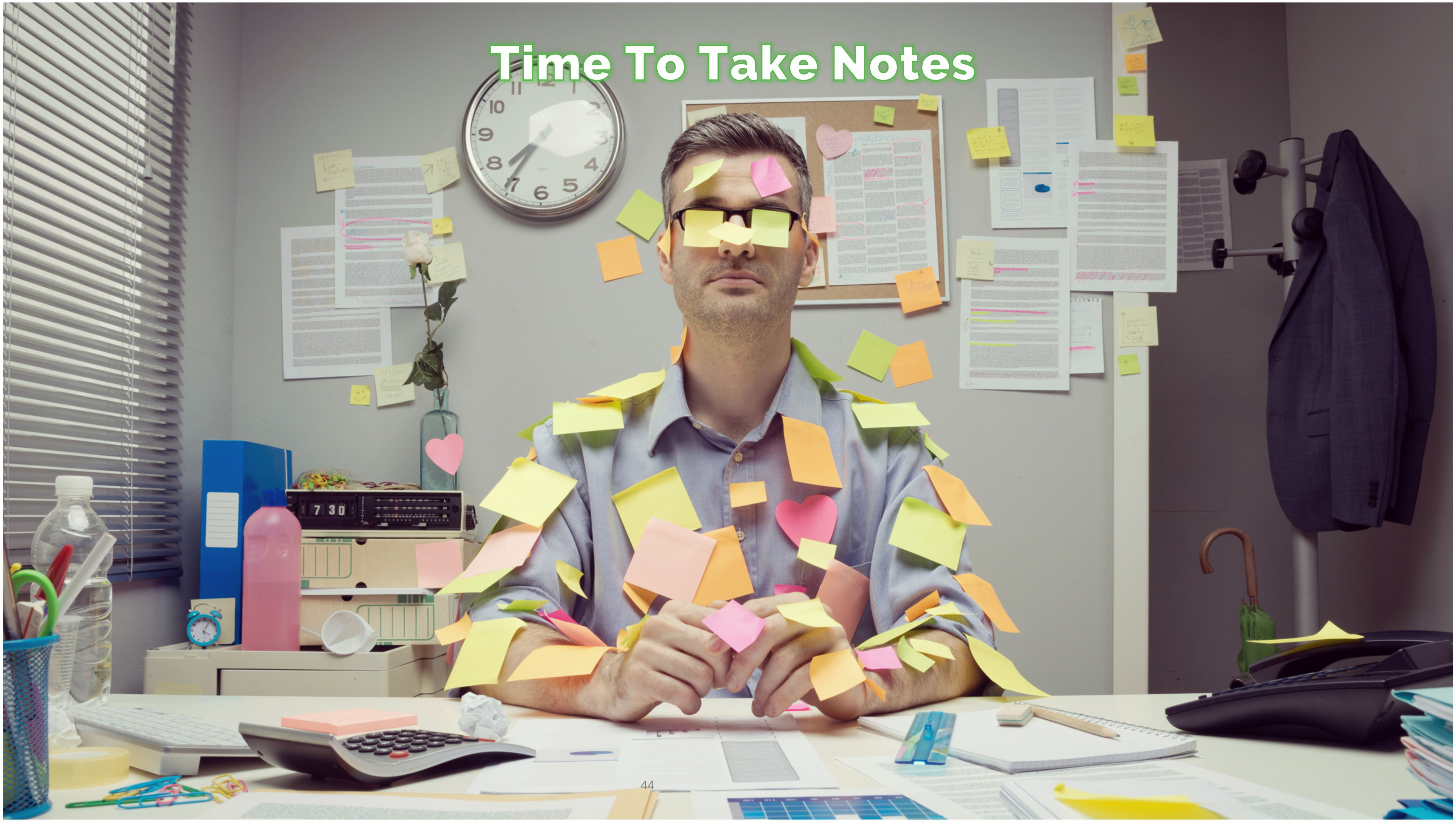
Partnerships

Additional Tips

- Securing 100 products takes years.
  - Start by investing 80% of the resources in 20% of the products.

- Reflect your success!
  - Trending charts of app sec metrics
  - Integration of tools into the build process
  - Share product certifications completion
  - Speak at Black Hat ☺

Time To Take Notes



Apply What You Have Learned Today

- Next week you should:
  - Generate security engagement questionnaire (10 Yes/No Qs)
  - Identify security tool implementation quick wins
- In the first three months following this presentation you should:
  - Establish an application security maturity program
  - Develop a product security strategy based on
    - Company's strategy
    - Development methodologies & pipelining tools
    - Product Types
- Within six months you should:
  - Hopefully map all products & owners ☺
  - Start executing the strategy

THANK YOU

Nir Valtman

@ValtmaNir