START WITH SECURITY

AUSTIN

#StartwithSecurity

Welcome

Bobby Chesney
Director, Robert Strauss Center for International Security and Law, University of Texas

Introductory Remarks

Dama Brown
Regional Director, Federal Trade Commission

Opening Remarks

Terrell McSweeny
Commissioner, Federal Trade Commission

Panel 1: Starting up Security

Building a Security Culture

Featuring

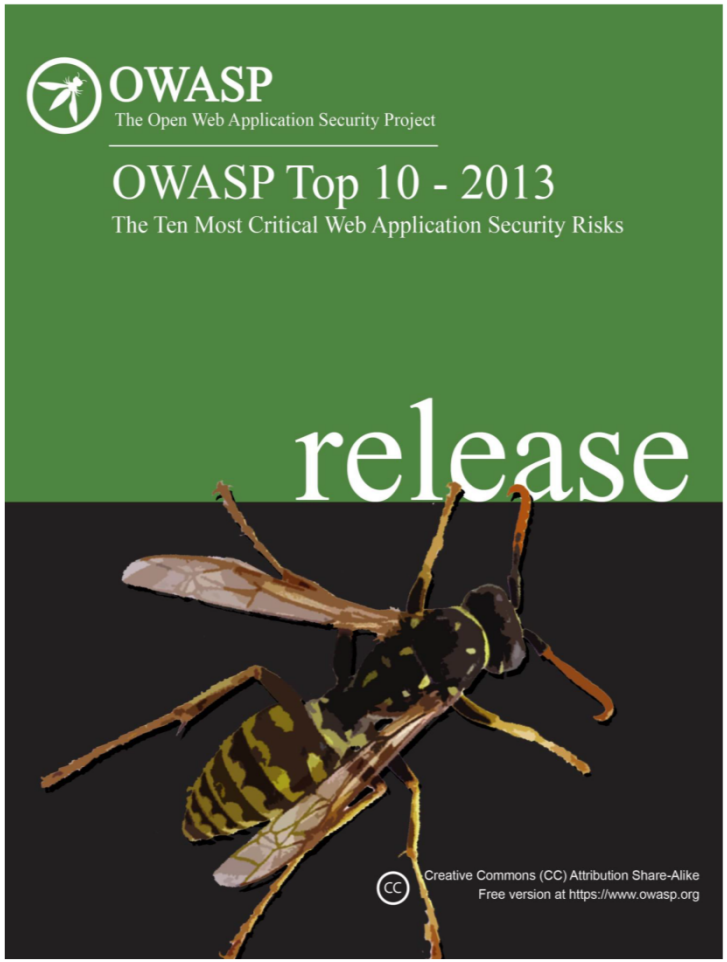
- **Christophe Borg**, VP Engineering Operations, RetailMeNot
- **Alan Daines**, Chief Information Security Officer, Dell
- **Josh Sokol**, Information Security Owner, National Instruments

- Moderator: **Laura Riposo VanDruff**, Division of Privacy and Identity Protection, FTC

Building a Security Culture

- **Security as Core Value**
  - Founders, executives, and employees
- **Building Security Expertise**
  - Engineers with interest can become security champions
- **Leveraging the Security Community**
  - OWASP, BSides, $(ISC)^2$, ISSA, SANS, and other free and proprietary resources

Common Vulnerabilities



1. Injection
2. Broken Authentication and Session Management
3. Cross-Site Scripting
4. Insecure Direct Object References
5. Security Misconfiguration
6. Sensitive Data Exposure
7. Missing Function Level Access Control
8. Cross-Site Request Forgery
9. Using Components with Known Vulnerabilities
10. Unvalidated Redirects and Forwards

Cross-Site Scripting (XSS)

- High-risk, easy to exploit vulnerability
  - Present in 7 out of 10 web applications
  - Vulnerability not affected by language choice
  - Attacker can run JavaScript in victim's browser
- Consequences for company, consumers
  - Risks consumers' personal information through malware, adware, spyware
  - Reputation and other harms to company

Training to Prevent XSS

- Look for places where user input is displayed back on a web page
- Pass in html tags to see if you can inject special characters
- Consult OWASP XSS Prevention Cheat Sheet and other resources

Building a Security Culture

- **Security as Core Value**
  - Founders, executives, and employees
- **Building Security Expertise**
  - Engineers with interest can become security champions
- **Leveraging the Security Community**
  - OWASP, BSides, $(ISC)^2$, ISSA, SANS, and other free and proprietary resources
- **Integrating Threat Modeling**
  - Consider potential threats early
- **Using Secure Frameworks**
  - Don’t reinvent the wheel

BREAK

#StartwithSecurity

Panel 2: Scaling Security

Adapting Security Testing for DevOps and Hyper-growth

Featuring

- **Matt Johansen**, Directory of Security, Honest Dollar
- **Matt Tesauro**, Senior Software Security Engineer, Pearson
- **James Wickett**, Engineer of Awesome, Signal Sciences Corp.

- Moderator: **Laura Berger**, Division of Privacy and Identity Protection, FTC

Vulnerabilities are Everywhere

Vulnerability percentage class by language

	ASP	Coldfusion	.NET	Java	Perl	PHP	Ruby
Cross-Site Scripting	49	46	35	57	67	56	29
Information Leakage	29	24	44	15	11	17	55
Content Spoofing	5	4	5	8	6	7	3
SQL Injection	8	11	6	1	3	6	-
Cross-Site Request Forgery	2	2	2	4	4	2	-
Insufficient Transport Layer Protection	0.8	1	0.9	1	0.3	4	0.7
Abuse of Functionality	0.3	6	0.3	0.9	0.5	0.2	-
HTTP Response Splitting	0.9	3	0.8	2	0.8	0.3	-
Predictable Resource Location	0.1	0.1	0.0	0.2	0.1	1	6
Brute Force	0.7	0.3	1	2	0.8	1	-
URL Redirector Abuse	0.7	0.4	0.5	1	1	0.9	-
Insufficient Authorization	0.2	0.3	0.5	0.9	1	0.2	0.7
Fingerprinting	0.3	0.1	0.5	0.6	0.3	0.1	0.7
Session Fixation	0.2	0.3	0.2	0.6	0.1	0.3	-
Directory Indexing	-	-	0.0	0.0	-	0.3	

* Limited amount of data available

They get fixed slowly…

	ASP	Coldfusion	.NET	Java	Perl	PHP	Ruby
Cross-Site Scripting	254	220	238	235	265	111	233
Information Leakage	354	110	250	180	292	136	687
Content Spoofing	267	187	232	238	185	99	79
SQL Injection	206	211	142	118	66	75	–
Cross-Site Request Forgery	226	195	207	209	76	114	–
Insufficient Transport Layer Protection	94	95	83	64	97	111	–
Abuse of Functionality	321	283	243	233	174	171	–
HTTP Response Splitting	751	188	228	73	155	27	–
Predictable Resource Location	674	53*	328*	120	3*	73	2
Brute Force	247	239	174	183	344*	149	–
URL Redirector Abuse	189	104	148	113	115	141	–
Insufficient Authorization	182	21	198	99	89	143	–
Fingerprinting	144	202*	155	118	0*	63*	–
Session Fixation	240	119	255	143	–	117	–
Directory Indexing	–	–	0*	41*	–	119	–

* Limited amount of data available

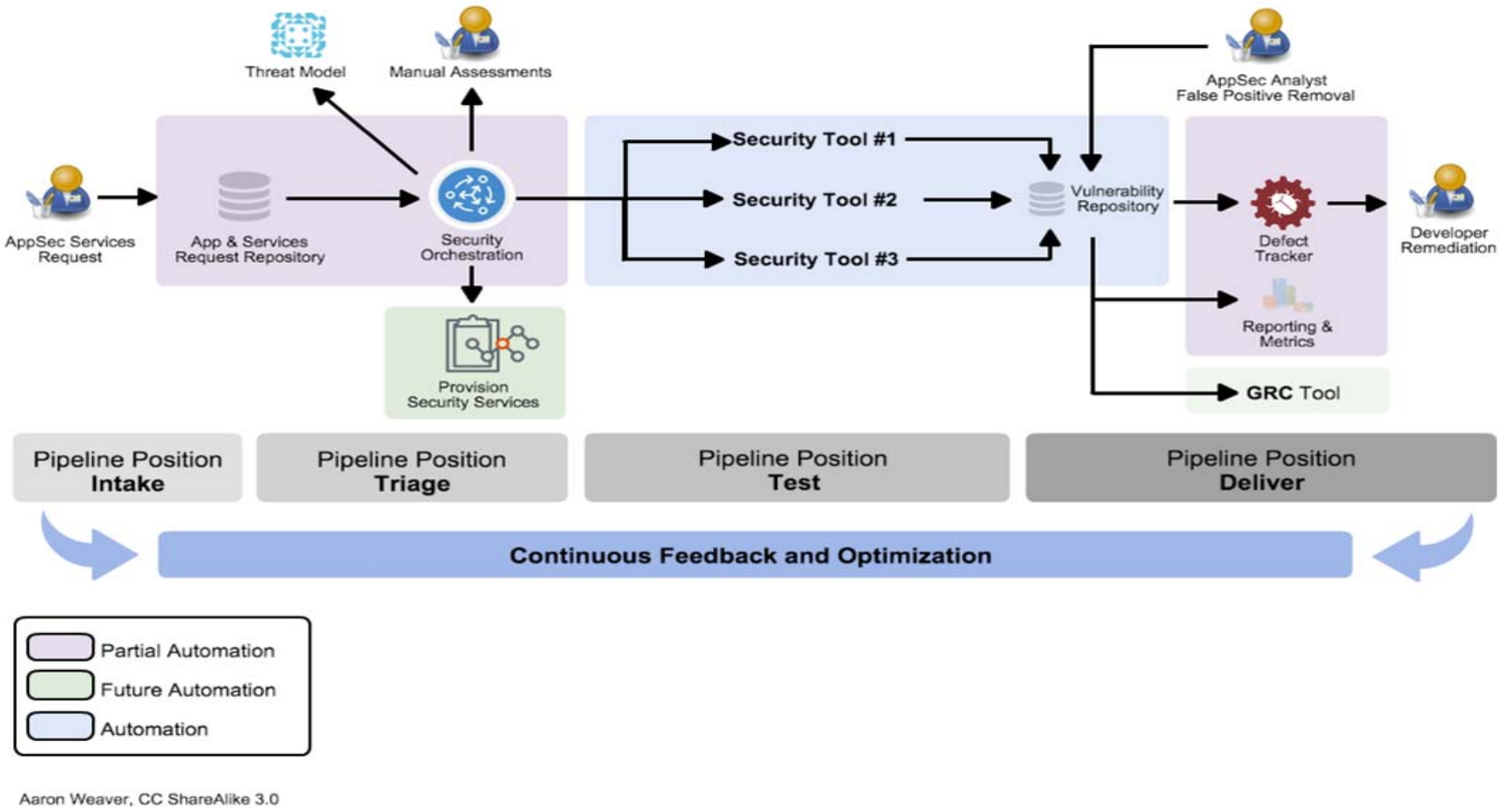
...if at all

Remediation percent by vulnerability class

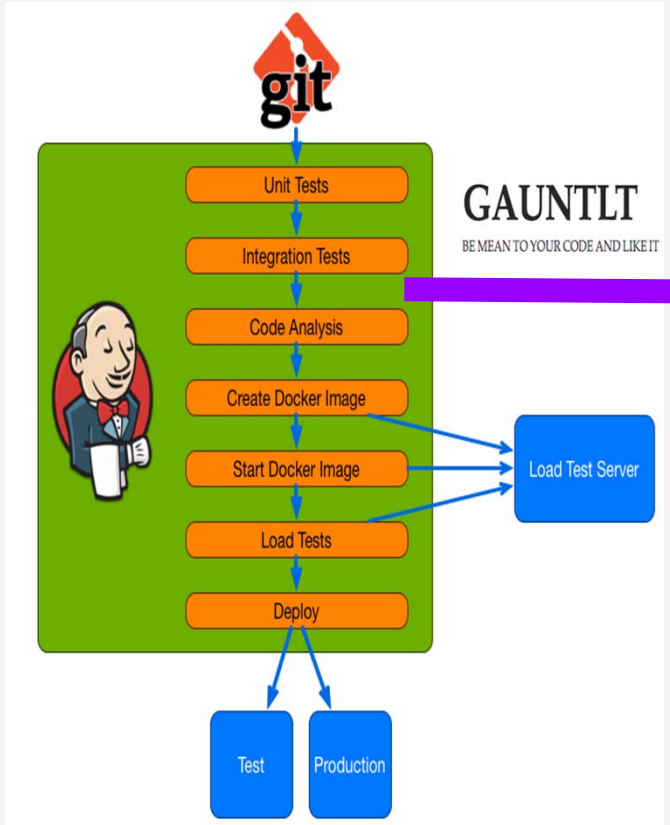
	ASP	Coldfusion	.NET	Java	Perl	PHP	Ruby
Cross-Site Scripting	79	75	76	71	85	65	67
Information Leakage	67	60	72	51	24	36	39
Content Spoofing	74	77	74	74	84	55	80*
SQL Injection	87	96	89	89	18	25	–
Cross-Site Request Forgery	60	46	54	51	69	54	–
Insufficient Transport Layer Protection	50	87	46	51	100	52	0*
Abuse of Functionality	62	100	62	78	70	65	–
HTTP Response Splitting	80	80	51	40	40	75	–
Predictable Resource Location	71	50*	38	22	100*	67	100
Brute Force	34	62	41	25	19	37	–
URL Redirector Abuse	44	53	49	66	26	32	–
Insufficient Authorization	80	80	83	51	32	50	0*
Fingerprinting	33	67*	41	56	20*	33	0*
Session Fixation	45	42	49	34	0*	48	–
Directory Indexing	–	–	100*	100*	–	100	–

* Limited amount of data available

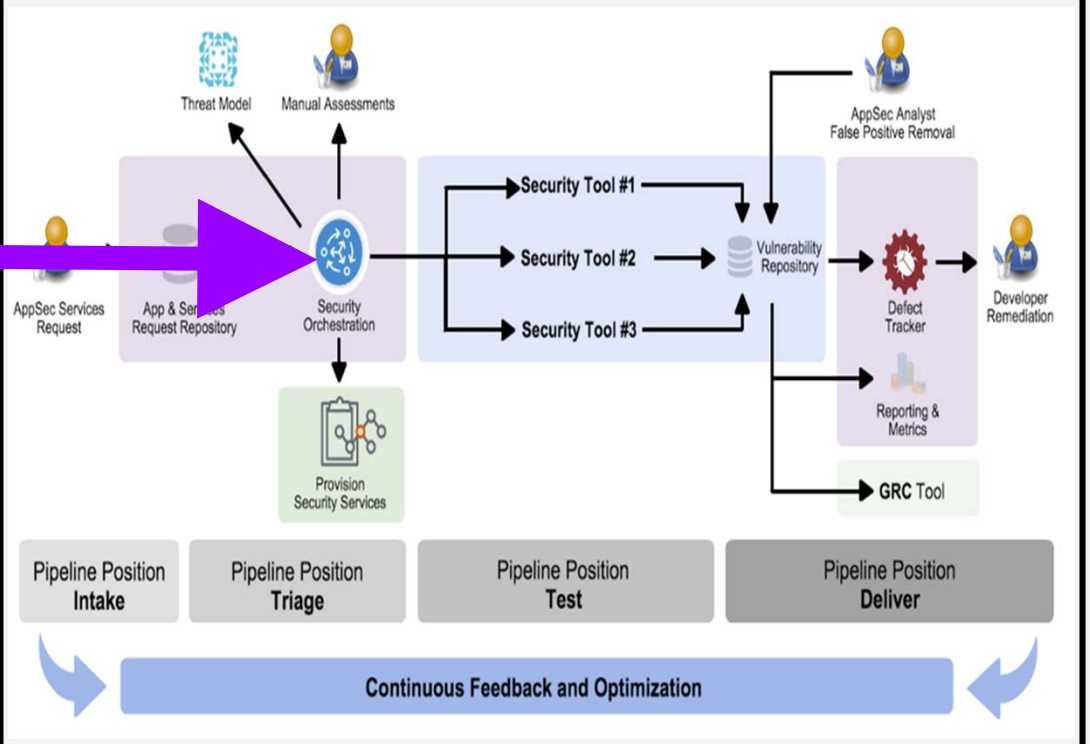
Scaling Security: a Rugged DevOps AppSec Pipeline Template



Integrating into the DevOps Pipeline



DevOps Pipeline



AppSec Pipeline

Gauntlt Example

```
Feature: nmap attacks for example.com
  Background:
    Given "nmap" is installed
    And the following profile:
      | name            | value       |
      | hostname        | example.com |

  Scenario: Verify server is open on expected ports
    When I launch an "nmap" attack with:
      """
      nmap -F <hostname>
      """
    Then the output should contain:
      """
      80/tcp  open  http
      """
  Scenario: Verify that there are no unexpected ports open
    When I launch an "nmap" attack with:
      """
      nmap -F <hostname>
      """
    Then the output should not contain:
      """
      25/tcp
      """
```

LUNCH

#StartwithSecurity

Investing in Security:

Fireside Chat with LiveOak Venture Partners Co-founder Venu Shamapant

Moderated by Commissioner Terrell McSweeny

Panel 3: Third-party AppSec

Dealing with Bugs, Bug Reports, and Third-party Code

Featuring

- **HD Moore**, Chief Research Officer, Rapid 7
- **Katie Moussouris**, Chief Policy Officer, HackerOne
- **Wendy Nather**, Research Director, Retail Cyber Intelligence Sharing Center

- Moderator: **Jarad Brown**, Division of Privacy and Identity Protection, FTC

Managing Third-party Software Security

- **More information**
  - Third Party Software Security Working Group, *Appropriate Software Security Control Types for Third Party Service and Product Providers,* http://docs.ismgcorp.com/files/external/WP_FSISAC_Third_Party_Software_Security_Working_Group.pdf

Managing Service Provider and Vendor Security

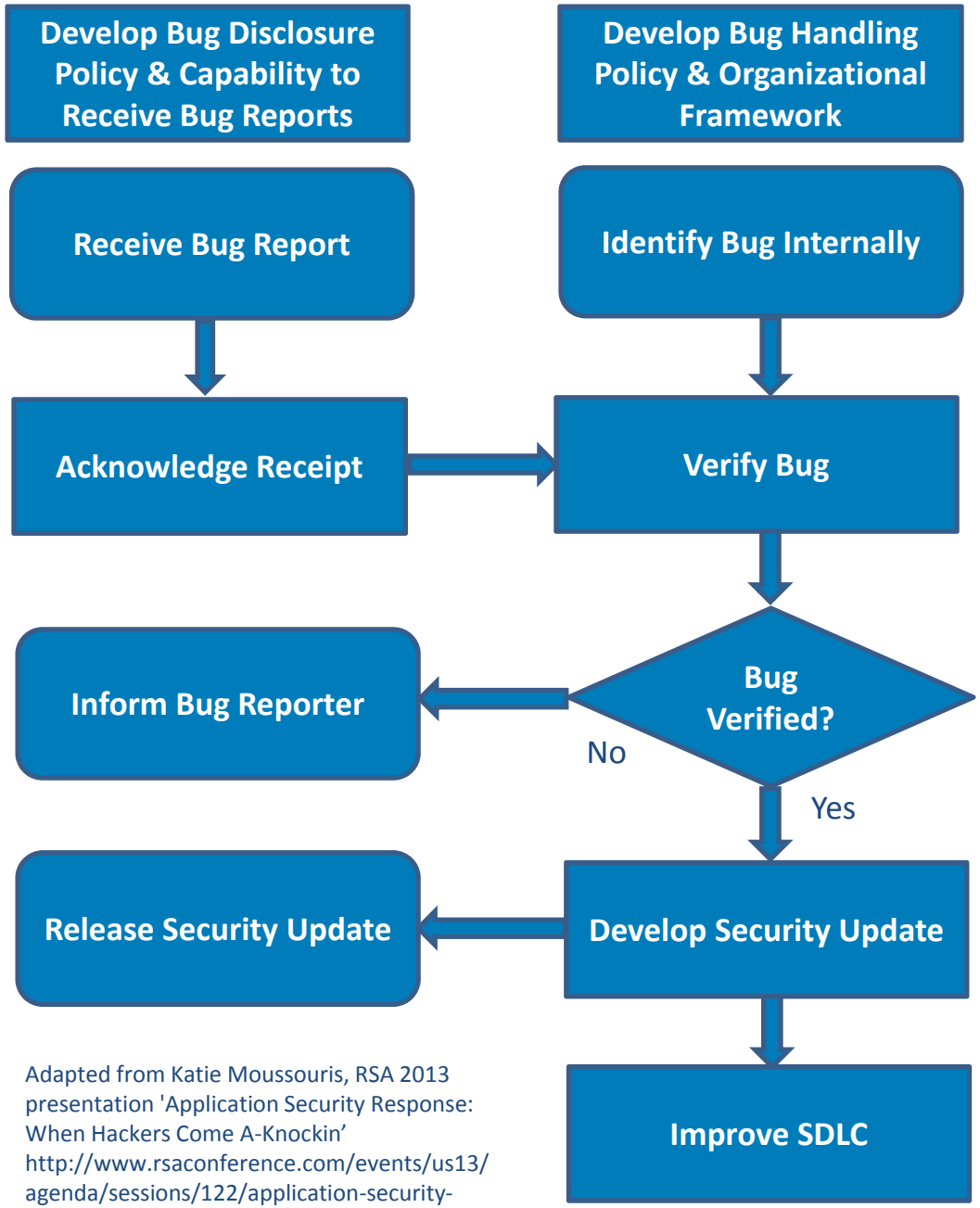
- **Evaluating Vendors**
  - Standard Information Gathering Questionnaire, https://sharedassessments.org/
  - Cloud Security Alliance Consensus Assessments Working Group Questionnaire, https://cloudsecurityalliance.org/group/consensus-assessments/
  - OWASP Secure Software Contract Annex, https://www.owasp.org/index.php/OWASP_Secure_Software_Contract_Annex

Managing Vulnerability Reports

- **Roll out the red carpet**
  - security@company.com
  - abuse@company.com
  - company.com/security
- **Process to Verify Reports**
- **Process to Address Reports**

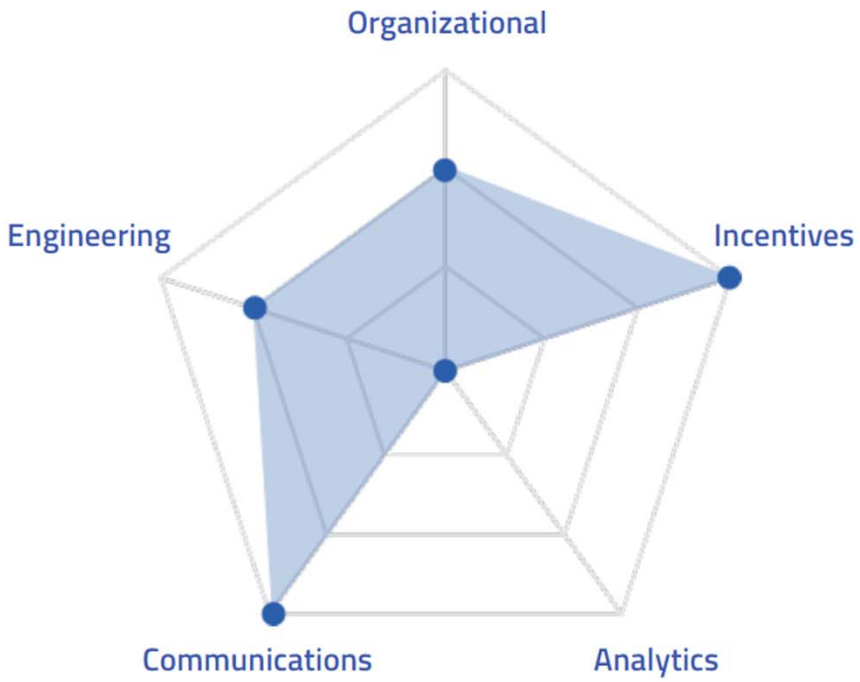
Disclosure

Response



Adapted from Katie Moussouris, RSA 2013 presentation 'Application Security Response: When Hackers Come A-Knockin'
http://www.rsaconference.com/events/us13/agenda/sessions/122/application-security-response-when-hackers-come-a

Vulnerability Coordination Maturity Model



- New model for organizations to assess maturity of their vulnerability coordination process
- Model guides how to organize and improve efforts inside and outside of an organization
- 5 Capability Areas: Organizational, Engineering, Communications, Analytics and Incentives
- 3 Maturity Levels for each Capability: Basic, Advanced or Expert
- Companies can benchmark their capabilities against the industry

Organizational

People, process, and resources to handle potential vulnerabilities.

Level	Capability
Basic	Executive support to respond to vulnerability reports and a commitment to security and quality as core organizational values.
Advanced	Policy and process for addressing vulnerabilities according to ISO 29147 and ISO 30111, or a comparable framework.
Expert	You have executive support, processes, budget and dedicated personnel for handling vulnerability reports.

Engineering

Capabilities to evaluate and remediate security holes, and improve software development lifecycle.

Level	Capability
Basic	Clear way to receive vulnerability reports, and an internal bug database to track them to resolution. See ISO 29147.
Advanced	Dedicated security bug tracking and documentation of security decisions, deferrals, and trade-offs.
Expert	Use vulnerability trends and root cause analysis to eliminate entire classes of vulnerabilities. See ISOs 29147, 30111, 27034.

Communications

Ability to communicate to audiences internally and externally about vulnerabilities.

Level	Capability
Basic	Ability to receive vulnerability reports and a verifiable channel to distribute advisories to affected parties. See ISO 29147.
Advanced	Tailored, repeatable communications for each audience, including security researchers, partners, customers, and media.
Expert	Structured information sharing programs with coordinated distribution of remediation.

Analytics

Data analysis of vulnerabilities to identify trends and improve processes.

Level	Capability
Basic	Track the number and severity of vulnerabilities over time to measure improvements in code quality.
Advanced	Use root causes analysis to feed back into your software development lifecycle. See ISOs 29147, 30111, 27034.
Expert	Track real-time telemetry of active exploitation to drive dynamic pivots of remediation strategy.

Incentives

Ability to encourage security researchers to report vulnerabilities directly.

Level	Capability
Basic	Show thanks or give swag. Clearly state that no legal action will be taken against researchers who report bugs.
Advanced	Give financial rewards or bug bounties to encourage reporting the most serious vulnerabilities.
Expert	Understand adversary behavior and vulnerability markets, and structure advanced incentives to disrupt them.

B R E A K

#StartwithSecurity

Panel 4: Beyond Bugs

Embracing Security Features

Featuring

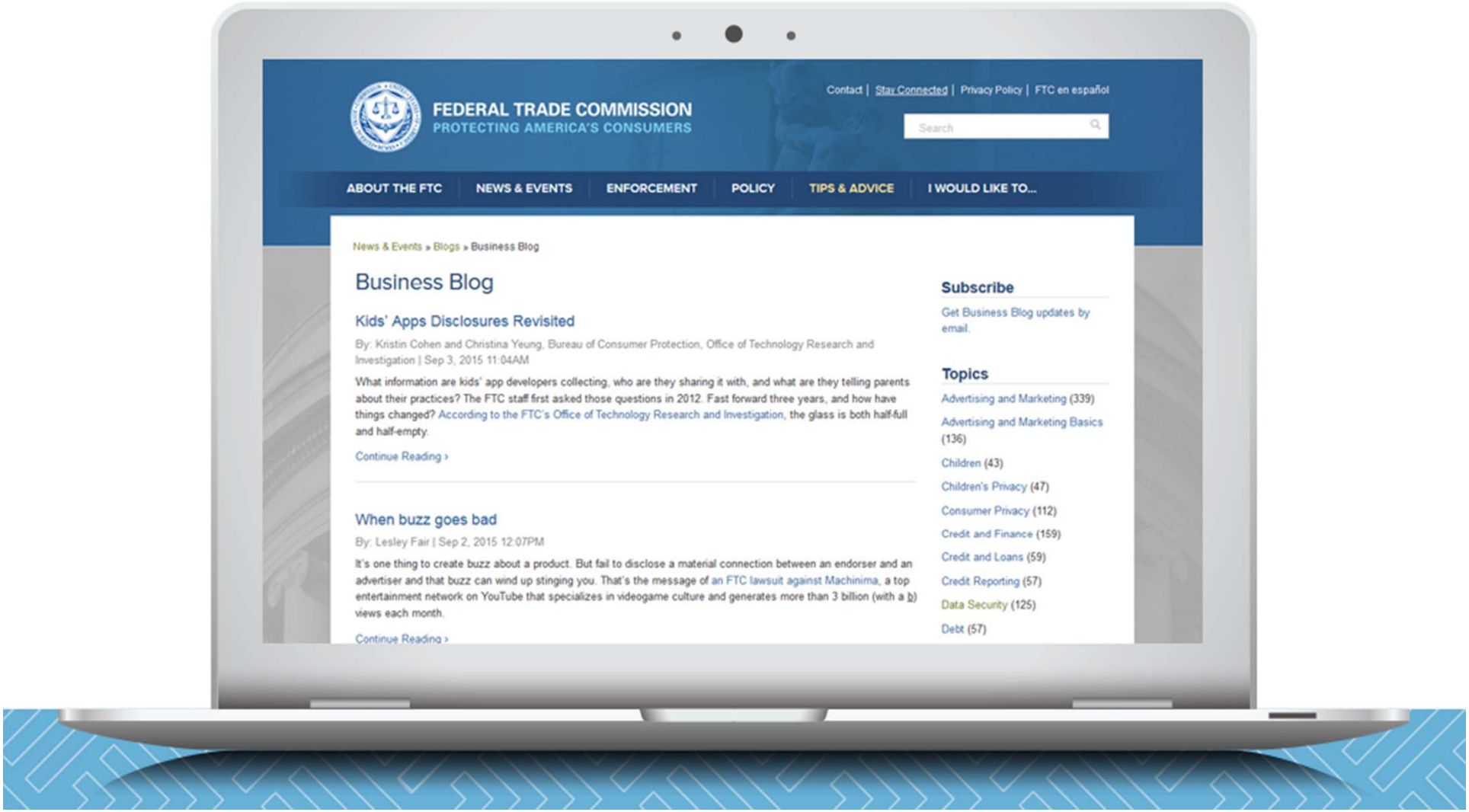
- **Robert Hansen**, VP of White Hat Labs, White Hat Security
- **Clare Nelson**, CEO, ClearMark Consulting
- **Caleb Queern**, Manager, KPMG Cyber

- Moderator: **Katherine McCarron**, Division of Privacy and Identity Protection, FTC

T H A N K S !

Subscribe to the FTC Business Blog

business.ftc.gov/blog



ftc.gov/datasecurity

T H A N K S !