Stuck in the Middle with You

Security in Communications Networks

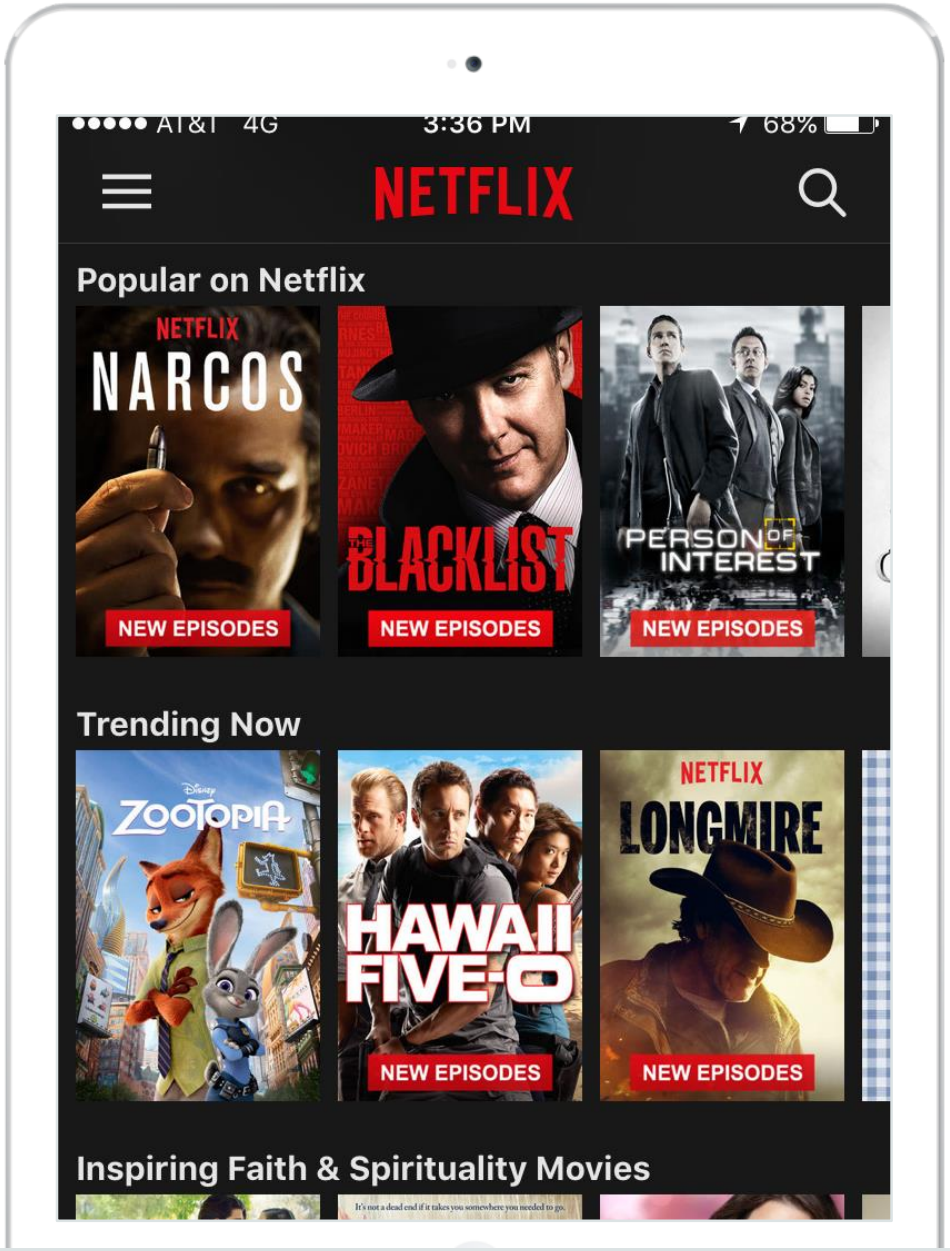
Jim Peterman
Director, Product Security
Oracle CGBU

Micah Williams
Senior Software Security Engineer
Oracle CGBU

2016 UNCC Cyber Security Symposium

We view the world as a hyper connected network of people, applications, and devices.

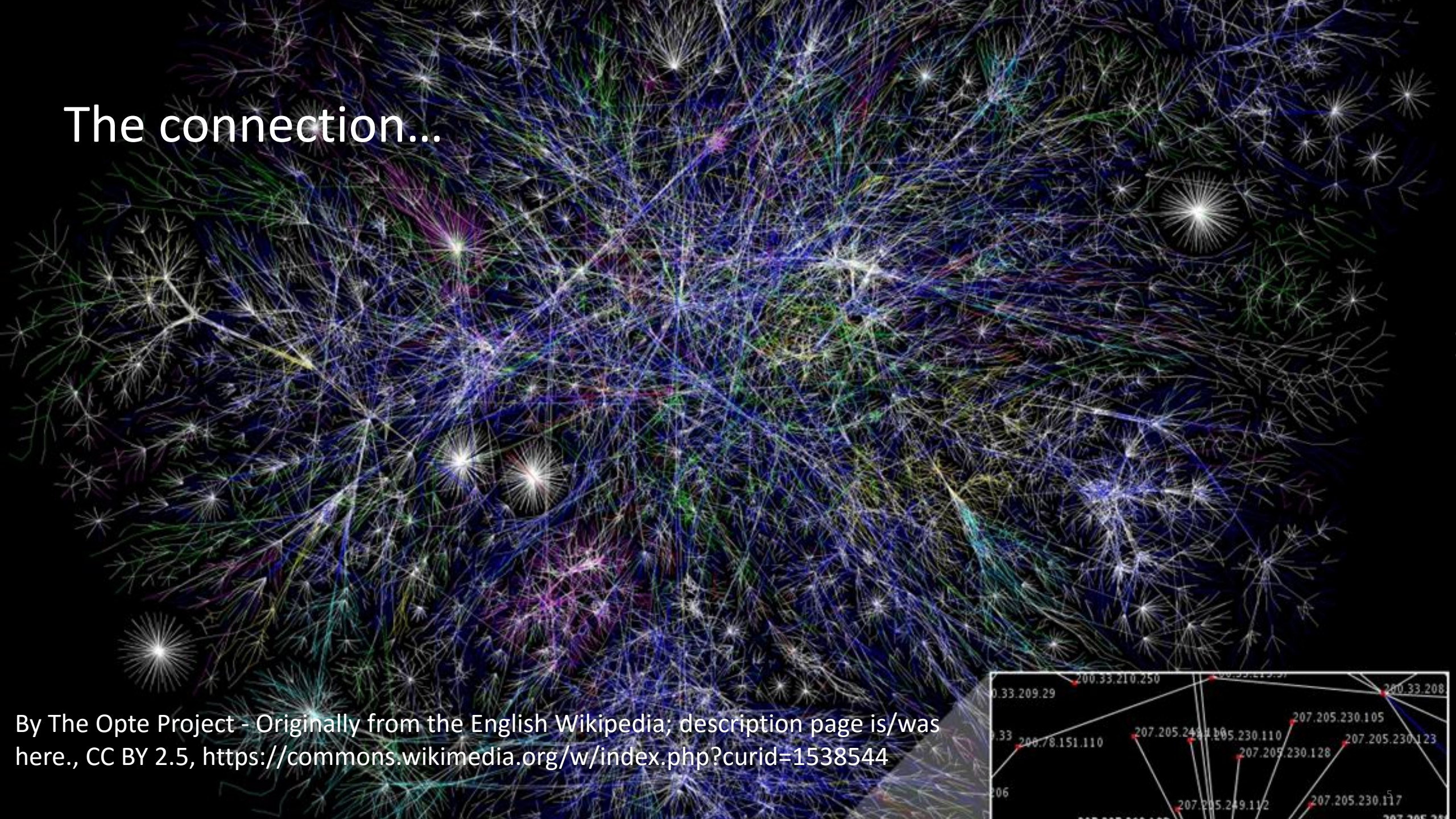
Connectivity…



User Expectations

- Confidentiality
  - "Need to know."
  - Secrecy and privacy is enforced to prevent unauthorized disclosure
- Integrity
  - "This must be right!"
  - Assurance of the accuracy and reliability of information and systems
  - Unauthorized modification is prevented
- Availability
  - "I need my data!"
  - Ensure reliability and timely access to data and resources to authorized individuals

The connection…



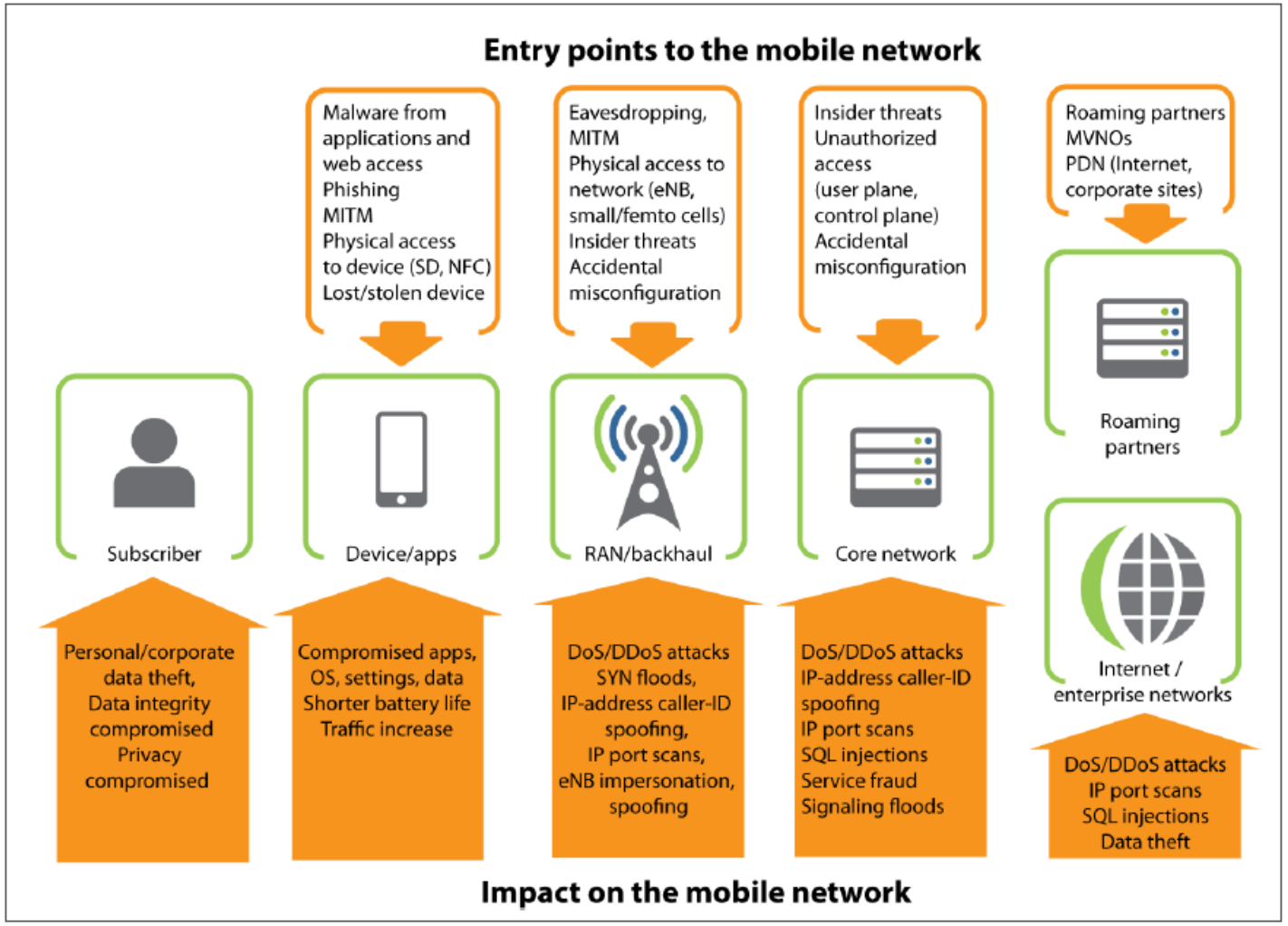
By The Opte Project - Originally from the English Wikipedia; description page is/was here., CC BY 2.5, https://commons.wikimedia.org/w/index.php?curid=1538544

The network in between…

- Attributes
  - Access control
  - Authentication
  - Session management
  - Secure transmission
  - Multi-session capability
  - Roaming
  - Accounting

- What makes it work?
  - Sophisticated SW controlled network elements
  - Complex comm protocols
  - Signaling protocols
  - Databases managing subscriber profiles
  - Policy engines
  - Interoperability among thousands of vendors

Network Threats



- Numerous entry points provide more opportunities for attackers.
  - Malware on handsets
  - Wi-fi alternate access methods
  - Roaming partner connections
  - End-to-end IP
    - DoS/DDoS
    - IP spoofing
    - IP port scanning

Note: Figure from Senza Fili Consulting

Critical Concerns…

- Attacks on networks and enterprises continue to escalate.
- Attacks are growing in frequency and sophistication.
- Addressing the attacks requires collaboration with suppliers.

The solution begins with…

…design; we must equip developers to write code securely.

Security In the Release Cycle

Security Release Criteria

Establish clear release criteria that drives OSSA compliance for:

- SCS adherence and validation
- Static Analysis
- Dynamic Analysis and Fuzzing

Bug backlog

- Review all open security vul'n and compliance bugs.
- Fix as many as possible in this release by severity and age.

3rd Party Management

- Uplift all 3rd party to latest release.
- Prioritize removal of unsupported items.

Architectural Risk Analysis

- Develop/maintain the ARA threat model.
- Assess threats and security controls across all I/Fs and assets.
- Carefully examine trust boundaries.
- Open bugs for any gaps where threats are not fully mitigated.
- Fix all critical/high findings

Apply Secure Coding Stds

- Follow all applicable SCS & validate via unit test and peer review.

Vulnerability Scans

- Routinely execute security vul'n scans & apply 3rd party security refreshes.

Static Code Analysis

- Apply static code analysis tools with each build, as practical, to improve code.

Update Security Guide

Code Analysis

- Complete static & dynamic analysis with approved tools such as HP Fortify and WebInspect.
- Assess findings and fix critical/high findings.

Fuzz Testing

- Test all exposed I/Fs & protocols identified in ARA

Other

- Assess/update TLS ciphers
- 3rd party security refresh

PSC
RSC
Bug status
3rd Party status

Legend
GA: release date
R-X: days prior to milestone

The solution ends with…

…partnership; suppliers and customers working together.

Customer – Supplier Partnership

- Clear communication
- Careful sharing of information
- Exchange of ideas
- Collaborative response

"It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness…"

– Charles Dickens, A Tale of Two Cities

By Jeremiah Gurney - Heritage Auction Gallery, Public Domain, https://commons.wikimedia.org/w/index.php?curid=8451549

Findings…

- Reduce security vulnerabilities in code
- Find security weaknesses and vulnerabilities internally
- Fix them before anyone else finds them.

From Strategy to Practice

The Customer's View

Who breaks into Telecom Systems?

- State-Funded Professionals
- Developers of "Hacker Toolkits"
- Insiders
- Security Researchers

No CVE ? No Problem.

- Malicious and "grey hat" hacker communities contain information on vulnerabilities; Some groups are focused specifically Telecom systems.
- Even worse: "0-Days" that are not patched and have no public disclosure.
- Useful to customers, but also to attackers.

NO NEWS ISN'T NECESSARILY GOOD NEWS

…In such an unforgiving threat landscape, Telecom customers *need* a partner in security.

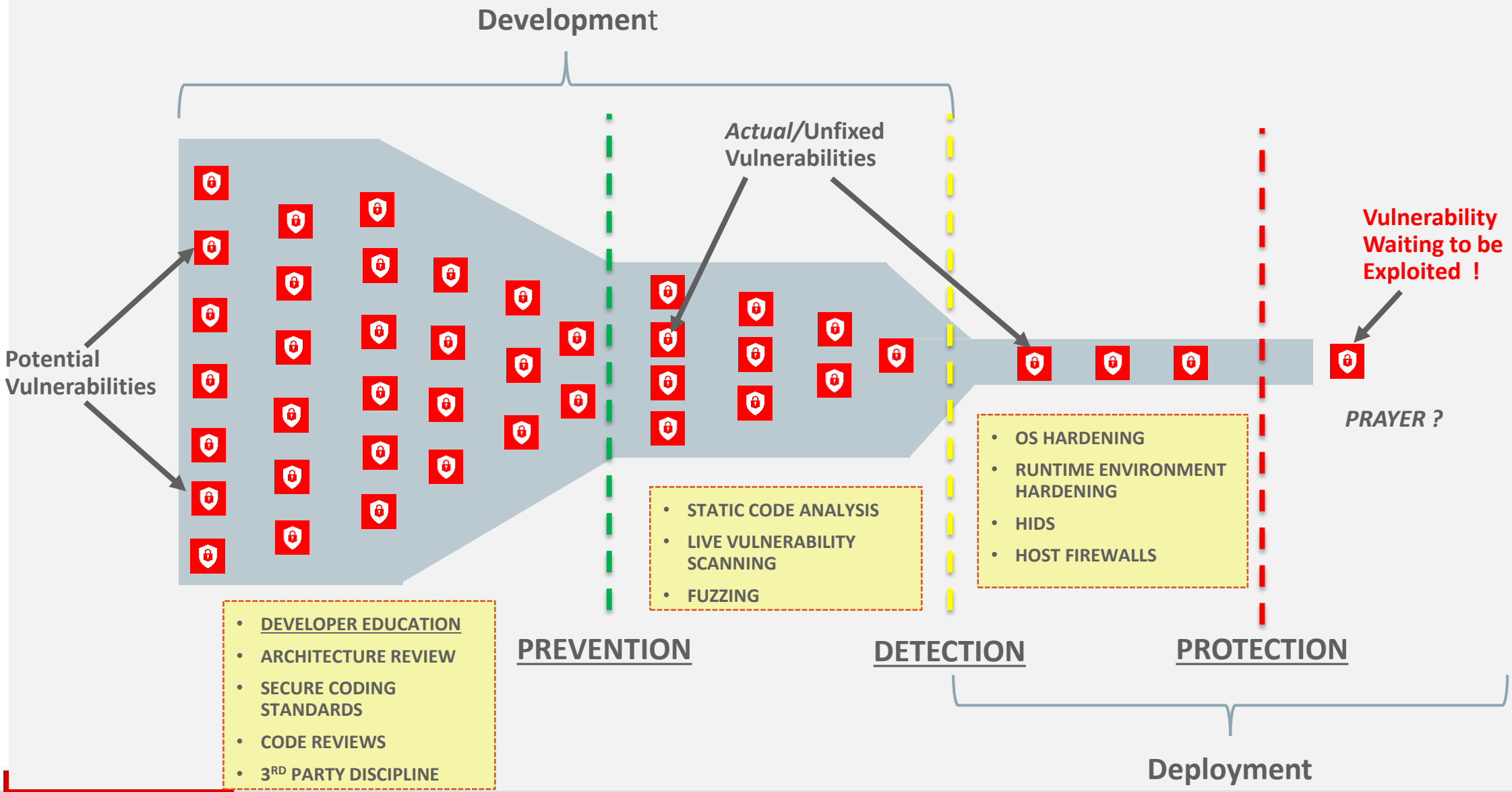
What customers expect of *us*

- Secure out of the box
- Flexible security configuration
- Ship code that is hardened and assumes it will be run in a "hostile" environment.
- Provide adequate logging of security-related events.
- Alert when product is configured in an insecure way.

You can trust products when …

- Product design incorporates security
- Developers comply with documented secure coding standards
- 3rd party software is carefully evaluated before being included into any release
- All source code (incl. 3rd party code) is run through static code analysis that focuses on security vulnerabilities.
- Security-focused dynamic testing such as fuzzing is applied to every release

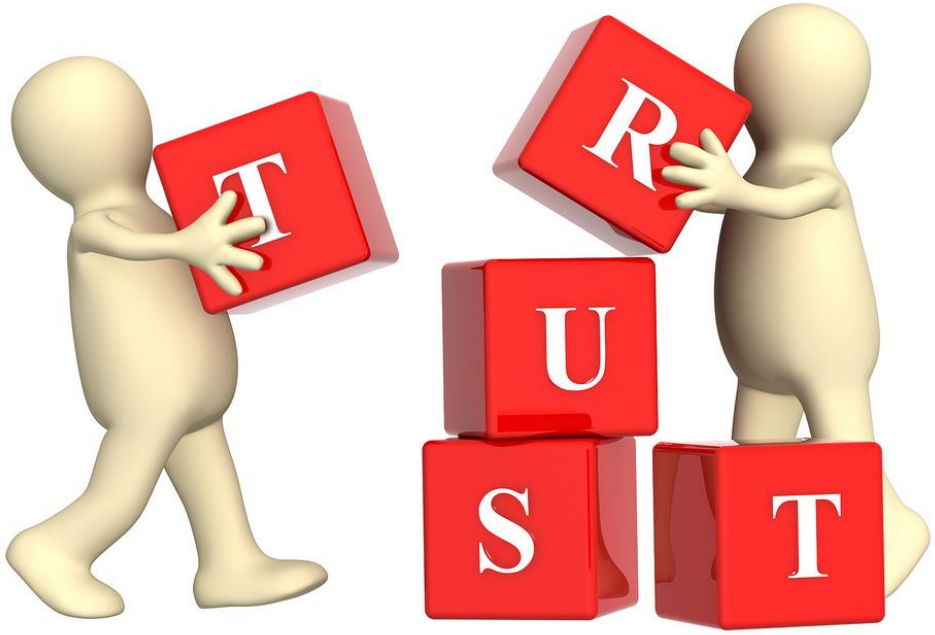
Software Vulnerability Prevention from Dev. to Live



Trust, but not blindly

Customers test too …

- Commercial fuzzing tools
  - Defensics, Peachfuzz
- Vulnerability scanners
  - Nessus, Qualys, Mcafee, Rapid7
- In-house penetration testing teams
- External pen-testing by 3rd parties



Real Life

What happens when…

- A major service provider customer call with an urgent finding.
- **Incident**: Internal penetration test reveals security vulnerability.
- Report shared privately with you (Responsible disclosure)
- Rollout of product halted until response is received.
- Customer deeply concerned since earlier version of product is deployed in network.

Case Study : Internal Response – Week 1

- Security Lead shares report with internal security research and ethical hacking team.
- Internal software security team digs through report:
  - Can issue be reproduced internally?
  - What is the root cause of the issue ?
  - Are earlier releases also vulnerable ?
  - Are there any workarounds or mitigations ?
  - Do we agree with the customer's assessment of the severity of the issue?
- Meanwhile, high-level management engage with customer via phone calls and email.

Case Study : Internal Response – Week 1 , Continued

- Within 48 hours, security team has released first technical response for internal use:
  - We'd seen this before :
    (Oh OK, It's THAT thing…… ☺)
  - Bug already discovered by the development team's early efforts at fuzzing.
  - Fix was already applied for a future release, but discovery was too late for product already out the door.
  - Severity at time not considered high enough for emergency patch.
  - Exploit of vulnerability as described by customer was possible, but only with additional private information . (A.C. was not as low as they claimed)
- Meetings scheduled for following week with customer IT officials and customer pen-test team.

Case Study : Customer Engagement - Week 2

- Management talks with customer network operations:
  - We acknowledge vulnerability, but disagree with the severity.
  - Plan is worked out for including fix in an emergency maintenance release.
  - Their pen-test team will conference with our product security team so we can describe justification for lowering CVSS.
  - On-site meeting at customer HQ is conducted.
- Product security team conferences with customer pen-test team
  - Technical details discussed and verified
  - We “haggle” over the CVSS score and agree on a lower rating.

Case Study : The Way Forward – Week2 and Beyond

- Timelines for patch are discussed
- Regular status meetings set up going forward to inform customer of our testing on fix.
- Details worked out on how and when to publicly disclose the vulnerability.
- Product security works with development team for product to verify fix for vulnerability.
- Customer receives maintenance release and eventually upgrades existing installations with patched code.

Takeaways

- Respond quickly and demonstrate competence and concern.
- Establish meeting of teams.
- Foster collaboration and open sharing of information.
- Apply comprehensive test techniques, such as fuzz testing.
- Recognize the gap between test environments and the real world.

"In today's complicated, cyber-security hyper-threatened environment – trust boundaries don't exist … but you must TRUST your supplier."

– James Peterman, Oracle

Integrated Cloud

Applications & Platform Services


ORACLE®