Application Security in an Ever Changing Digital Landscape

Trace Hollifield
Security Architect
CISSP, GSLC, CISM, C|CISO, ITIL
Trace.Hollifield@microfocus.com
Agenda

- Security Landscape
- Specific Application Security (App Sec) Challenges
- Establishing an AppSec Program
- Why AppSec is Hard
- Software Security Research Results
- Conclusion
Security Challenges
Breaches Around the world

July 2015

- **Anthem**: 80,000,000
- **Home Depot**: 80,000,000
- **Ebay**: 145,000,000
- **Target**: 70,000,000
- **UPS**: 2,000,000
- **Adobe**: 35,000,000
- **Living Social**: 50,000,000
- **Ubisoft**: "unknown"

**145 million people affected**

77 days time to detect

July 2017

- **Anthem**: 80,000,000
- **Friend Finder Network**: 402,000,000
- **Deep Root Analytics**: 338,000,000
- **Fling**: 400,000,000
- **HD Supply**: 1,000,000

**500 million people affected**

600 days time to detect

---

http://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/
90 percent of reported security incidents result from exploits against defects in the design or code of software. Ensuring software integrity is key to protecting the infrastructure from threats and vulnerabilities and reducing overall risk to cyber attacks. To ensure system reliability, integrity, and safety, it is critical that provisions be included for built-in security of the enabling software.
Got Vulnerabilities?

Since 1999, over 147k vulnerabilities have been registered.

41% of all vulnerabilities are from recent years.

7% are from 2019 (Only 63 days in)

Malware continues to grow and morph as new vulnerabilities are discovered and old vulnerabilities are not remediated.

• ~50 Million variants in the last 60 months

Malware – Attack Vectors and Cost

Malware Cost

Trojans:
- Keylogger Detective 2.3.2 (w/ hidden installation). Price: **US $3**
- Web Browser Based (Opera, Mozilla Firefox, Chrome, Safari, etc). Price: **US $8**
- Spider Keylogger Pro v. 1.2.4. Price: **US $50**

Ransomware:
- Winlocker Source Code. Price: **US $8**
- Winlocker Activation Key. Price: **US $10-20**

Source – TrendMicro “Russian Underground 101”

Malware Highlights
- **Low Cost** … The cost of various malware is extremely low for the possible greater rewards
- **Accuracy by Volume** … The attacker only needs to be right once while defense has to be right all the time
- **Go Where the People Are** … Web attacks are the most popular vector to infect hosts due to people’s habits, advertising conduits, and rapid changes in the environment which create challenges for data security defenders.
Ransomware Jackpot

FBI Received Over 2,600 Ransomware Complaints in 2016 Costing $1.3 B

• About 2673 complaints were submitted according to IC3’s report. The number is just the tip of the iceberg, though, when compared to the 298,728 cybercrime-related complaints received overall last year, 2016. Losses connected to such cybercriminal activities is reported to be around $1.3 B.

WannaCry: Encrypts 176 file types, including database, multimedia, and archive files, as well as Microsoft Office documents.

Petya: Encrypts the system’s files, overwrites its Master Boot Record, and locks users out with a blue screen of death.

Locky: Encrypts over 130 file types, including those on removable drives and unmapped network shares.

Hidden Tear: Is an open-source ransomware that allowed cybercriminals to create their own versions which were themselves reworked into more spinoffs. One variant can encrypt up to 2,783 file types.

Cerber: Sold as a ransomware as a service (RaaS), which means cybercriminals can customize its encryption behavior and ransom demands; it’s been recently spotted to be capable of stealing from Bitcoin wallets and evading machine learning.

Data Security Standards / Frameworks

- Several to pick from: ISO 27000, PCI-DSS, NIST, COBIT, SANS 20 and more
- Purpose of the Frameworks: To provide guidance on how to protect sensitive data
- Payment Card Industry – Data Security Standards (PCI-DSS) started in 2004 with the following requirements:

<table>
<thead>
<tr>
<th>Goals</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build &amp; Maintain a Secure Network</td>
<td>1. Install and maintain a firewall configuration to protect cardholder data</td>
</tr>
<tr>
<td></td>
<td>2. Do not use vendor-supplied defaults for system passwords and other</td>
</tr>
<tr>
<td></td>
<td>security parameters</td>
</tr>
<tr>
<td>Protect Cardholder Data</td>
<td>3. Protect stored cardholder data</td>
</tr>
<tr>
<td></td>
<td>4. Encrypt transmission of cardholder data across open, public networks</td>
</tr>
<tr>
<td>Maintain a Vulnerability Management Program</td>
<td>5. Use and regularly update anti-virus software or programs</td>
</tr>
<tr>
<td></td>
<td>6. Develop and maintain secure systems and applications</td>
</tr>
<tr>
<td>Implement Strong Access Control Measures</td>
<td>7. Restrict access to cardholder data by business need to know</td>
</tr>
<tr>
<td></td>
<td>8. Assign a unique ID to each person with computer access</td>
</tr>
<tr>
<td></td>
<td>9. Restrict physical access to cardholder data</td>
</tr>
<tr>
<td>Regularly Monitor and Test Networks</td>
<td>10. Track and monitor all access to network resources and cardholder data</td>
</tr>
<tr>
<td></td>
<td>11. Regularly test security systems and processes</td>
</tr>
<tr>
<td>Maintain an Information Security Policy</td>
<td>12. Maintain a policy that addresses information security for all personnel</td>
</tr>
</tbody>
</table>

https://www.pcisecuritystandards.org/document_library
Being fully PCI Compliant...

Key Points
• Not 100% across the board
• Compensating controls (duct tape) fill in the blanks
• Qualified Security Assessors (QSAs) work with shared data and limited time engagements
• Compliance testing is only once a year

But we are PCI compliant... Not So Much.

Large Misses
- Requirement 10: Logging and Log Review
- Requirement 1: Network Environment & Changes
- Requirement 3: Data at Rest Protections
- Requirement 6: Vulnerability Management & Patching
- Requirement 7: Need to Know Access & Permissions
- Requirement 12: Policies & Training

Challenge Summary

Bad Actors – External

• Deep Pockets & Cheap Tools
• Creative & Innovative
• Successful at extorting funds from a business

Bad Actors – Internal

• Many verticals lack compliance maturity
• Many verticals struggle with existing compliance requirements let alone new and evolving requirements
• Lack of structured Policies & Processes combined with enforcement
• Confirmed immaturity for Logging and Log Review, Data at Rest Protections, Vulnerability Management, Need to Know Access & Permissions
• Governments are jumping in on the action with new compliance regulations and fines

Business Impact

• Reputation and Confidence Loss
• Lawsuits
• Compliance Fines
• Combination of the three results in a decline of revenue while increasing business expenses
Application Security Challenges
Tsunami of Apps

1000 applications and counting...
Speed vs Depth

“I want 5 minute scans with no false positives.”
Developer User Story

We have seen the AppSec team
AND IT IS YOU! (the developer)
Establishing an AppSec Program
Goals and benefits of an Application Security Program

The mitigation of application security risks is not a one time exercise; rather it is an ongoing activity that requires paying close attention to emerging threats and planning ahead for the deployment of new security measures to mitigate these new threats. This includes the planning for the adoption of new application security activities, processes, controls and training.


A successful applications security programs need to:

- Map security priorities to business priorities
- Assess the current state and target state using a security program maturity model
- Seamlessly integrate into development processes and tool chains
Evolution of Capability for Application Security

Risk Identification
- Dynamic Security Testing
- Static Security Testing
- Regulatory Compliance

Policy Compliance
- Custom Reports
- Security Defect Prioritization
- Critical Defect Remediation
- Data Summary Reporting
- Initial Metrics

Risk Mitigation
- Continuous Delivery Integration
- Secure SDLC Monitoring
- Historical Correlation
- Custom Report Enhancement
- Security Defect Trending
- Release Management Integration
- Best Practices & Actionable Controls

Risk Prevention
- Education
- Awareness campaigns
- Continuous Testing
- Analysis in Depth
- Operational Metrics and KPIs
- Compliance enforcement
- Security Defect Eradication
- Secure Reusable Components
- Secure Repositories

Adaptive Protection
- Continuous Monitoring
- Runtime Protection
- Service Orchestration
- Adaptive Response
- Active Engagement
  (Proactive and Reactive)
Baseline and periodic maturity assessments key

<table>
<thead>
<tr>
<th>Blank Scorecard</th>
<th>Industry Best Practices</th>
<th>Enterprise Scoring</th>
<th>Prioritized Roadmap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objective 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objective 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objective 0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Education</th>
<th>Standard</th>
<th>Planning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Governance &amp; Alignment</td>
<td>Requirements &amp; Design</td>
<td>Verification &amp; Assessment</td>
</tr>
</tbody>
</table>

- Threat Mn | Sec Req | Def Design | Arch Rev | Code Rev | Sec Testing | Vul Mgmt | Infr Harden | Ops Enable |
- Education | Standard | Planning |
- Threat Mn | Sec Req | Def Design | Arch Rev | Code Rev | Sec Testing | Vul Mgmt | Infr Harden | Ops Enable |
Building Security Into the SDLC

Value of automated and manual analysis

**Static Analysis**
- 100% code coverage
- Pinpoint and prioritize violations within code authoring environment
- Root cause of vulnerabilities with line-of-code detail
- Language specific remediation strategies

**Dynamic Analysis**
- Stimulate application through automated, external security attacks
- Identifies, crawls and attacks application attack surface
- QA or production environments
- Doesn’t require code
- Real world attack simulation

**Manual Review**
- Threat modeling
- Requirements verification
- Security Architecture Reviews
- Business logic verification
- Reduce false positives
Building an AppSec Program – Major Milestones

- **Assess / Design**
  - Assess
  - Workshop
  - Baseline Assessment
  - Requirements
  - Data Readiness
  - Roadmap
  - Staff Planning
  - Role definition
  - Interviews
  - Hiring

- **Development**
  - Processes & Procedures
    - Business
    - Operational
    - Technical
    - Analytical
  - Architecture design
  - Analyst Development Plan
    - Hiring + Training
  - Deployment
    - Toolset Deployment
    - Toolset integration
  - Data Onboarding

- **Implementation**
  - Initial Monitoring Capability
    - Business Process Integration
      - Activate development and SDL
      - Rollout of Processes and Procedures (documentation and training)
    - Workflow Creation
    - Analyst Development Plan
      - Hiring + Training
    - Data Onboarding
  - Deployment
    - Toolset Deployment
    - Toolset integration

- **Mature / Operate**
  - Deliver KPIs and Metrics
  - AppSec Maturation
    - Enhancements, monitoring & system tuning, process and technology adjustments
  - Update Roadmap
    - Advanced Analysis
    - Integrated Hunt Operations
    - ML / Analytics
    - ODS

- **Operate / Transition**
  - Maturity Assessment

Source: Fortify Professional Services, 2018
Measure to demonstrate success

- % of security defects identified by sprint/phase
- % of security defects whose risk has been accepted vs. % fixed
- % of security defects per project over time (between quarter to quarter)
  - Vulnerability density (security defects/LOC)
- Average time required to fix/close security defects during design, coding, and testing
- Average time to fix security defects by defect type
- Average time to fix security defects by application size/code complexity
Why Application Security is Hard
## Development Point of View: Challenges/Concerns

<table>
<thead>
<tr>
<th>Security Gets Involved at Later Stages in the Dev Cycle</th>
<th>Full Scans Take Too Much Time!</th>
<th>Audit Process Takes Too Much Time!</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Traditionally, static or dynamic scans are run before releasing the app.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- This means either developers get dozens of issues to fix in a very short time or they’ll release the app with these issues.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- It is painful to go back to a project that you’ve already finished and fix things.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- When scans are initiated, developers don’t get results in days in some cases weeks.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Scanning the entire code base and auditing can take time.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Developers get security issues way later than they would like.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Auditing is still the #1 bottleneck for all application security efforts.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Even if scans are completed in minutes, human auditors work using FIFO queues and they’re outnumbered.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Audit results are challenged by developers and cause friction/time loss.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Where does Security fit in DevOps?
Promise vs Reality of Security in DevOps

Where does security currently fit?

99% of those surveyed agreed that DevOps is an opportunity to improve application security.

But only 20% perform application security testing during development. Most wait until late in the SDLC – or not at all!

Classic integration of security touchpoints can’t keep pace
DevOps Toolbox

AST Integration can be a challenge given tool diversity

IDE’s
- eclipse
- IntelliJ IDEA
- Visual Studio
- Team Foundation Server

Requirements & issues
- JIRA
- CA
- Rally
- JUnit
- Bamboo
- Team Foundation Server

Build servers & Build tools
- Bamboo
- TeamCity
- Jenkins
- PowerShell
- maven

Configuration automation
- Puppet
- Chef

Security tools including Open Source
- Fortify
- Black Duck
- Splunk
- Sonatype

Containers
- Docker

Communication/ChatOps
- HipChat
- SharePoint

Code repositories & apps
- Git
- Mercurial
- Bitbucket
- GitHub

Cloud
- Amazon Web Services
- EC2
- ServiceNow
- OpenStack
- Microsoft Azure
- Cloudera
Modern Application Security Programs Need to Adapt

Mobile Apps

Skills

Bimodal IT

Agility

Continuous Integration

DevOps

Continuous Development

Automation

Cloud

Open Source Software Components

Micro services and containers

Internet of Things

New languages

Software Supply Chain

More robust and dynamic apps

Microservices and containers

DevOps
Best practices for integrating security w/DevOps

Security should be part of the DNA of DevOps

• Make security part of the value stream
• Identify skilled early adopters
• Work in small consumable steps
• Standardize on toolset
• Early visible wins
• Focus more on the process than defect totals
• Begin with a loose security policy and tighten as process matures
• Mark builds as unstable but don’t fail builds until process is mature
Effective/High Velocity DevOps with Security built-in

Prerequisites:
1) Base line scan / audit performed
2) Confidence threshold defined
3) Audit Filters or Rules from previous scans applied via vuln mngt platform

IDE Security Plugin
Local Remediation Scan
Light Weight Static Scan

Developer

Download Artifact

Source Code Mgmt System

Vulnerability Mngt Platform

Defect Mgmt System

Lead Developer

New / Critical Issues Exist Alert

Triage & Assign to Defect Tracking System

Upload New Scan Merged with Existing

Developer is sent ticket with embedded link to issues

Check in Secure Code

Build / Translation Job

Translated Files sent to available scan server

Static Scan Servers Cluster

Security CAN be part of the DNA of DevOps
Software Security Research Results
Annual Application Security Research Report
Software Security Research team

Data from managed application security platform:
• Anonymized and sanitized vulnerability data collected over a year (Nov 2016 – Oct 2017)
• 7,800+ Web applications & 700+ mobile applications

Three themes from the 2018 AppSec risk report

1. Analysis shows broad vulnerability in apps
   The majority of web or mobile applications analyzed had at least one critical or high severity issue

2. OWASP Top 10 is a starting point
   1 out of 2 apps had critical or high vulnerabilities not covered by the OWASP Top 10 2017

3. GDPR is forcing strong protection of customer data
   GDPR strongly hints at the use of encryption and pseudonymization as acceptable approaches to protect personal data; applications are a potential weak link.
OWASP Top 10 is an industry best practice

OWASP Top 10 is a powerful awareness document for web application security.

- represents a consensus about the most critical security risks to web applications

Some standards reference OWASP Top 10:

- MITRE
- PCI DSS
- United States Federal Trade Commission

OWASP Top 10 is a starting point, but is not all-inclusive.

Many of the top reported security weaknesses in web application didn’t make the list, and it doesn’t include vulnerabilities to other attack surfaces of the organization.

90% of applications have at least one issue outside of the OWASP Top 10.
Categories outside of OWASP Top 10 (= None) are not lower in severity

1 out of 2 apps have critical or high vulnerabilities **not covered** by the OWASP Top 10 2017
Regulatory compliance with standards and best practices leave out issues across the board

Compliance with standards like OWASP Top 10, NIST, PCI DSS, or GDPR is a great place to start, but all standards have critical and high severity weaknesses issues that aren’t covered.

Depending on the standard, anywhere from 1/5 to 1/2 apps have critical or high vulnerabilities not covered by the regulatory mapping.
Summary
Making Application Security Matter

• Adapt and catch the wave!
• Establishing an application security capability is an evolutionary journey
• Modern application security programs need to adapt
• Security should be part of the DNA of DevOps
• OWASP Top 10 is a starting point for application security, but 1 out of 2 apps had critical or high vulnerabilities not covered by the OWASP Top 10 2017

2018 Application Security Research Update
Final thought
This stuff doesn’t happen overnight

Think big

Do small
Thank You.