Introduction

- Senior Manager @ Adobe’s Secure Software Engineering Team (ASSET)
- I lead the proactive security efforts.
- 🐦 @adobesecurity / @mohitkalra
Adobe’s Strategy

ADVANCING STATE OF THE ART FOR CONTENT

HARNESSING THE POWER OF DATA

DRIVING DIGITAL TRANSFORMATION OF INDUSTRIES
ADOBE CLOUD PLATFORM

Adobe Document Cloud  Adobe Creative Cloud  Adobe Marketing Cloud

CORE TECHNOLOGIES

CONTENT  DATA

PRIVATE, PUBLIC OR HYBRID CLOUD
Secure Product Lifecycle

Credit:
Secure Product Lifecycle (SPLC) is a set of processes designed to help product teams engineer secure software.
For our team, the approach to security is much more complex
Security is all about making choices
... and balance
Implementing security is about providing high ROI and business alignment
…. while trying to fix the weak links
The challenges in this complex world.
A central security team’s challenge #1

- Hiring skilled security professionals is difficult.
- Team needs to learn continuously.
- Time spent => high premium $$$.

Scaling the security work with a small team.
A central security team’s challenge #2

A growing and diverse company product portfolio.
A central security team’s challenge #3

The business critical products vs the legacy applications.
The challenges for a security team

- Security team’s bandwidth
- Diverse technology
- Varying business criticality
How can a security team overcome these challenges?
The challenges for a security team

- Security team’s bandwidth
- Diverse technology
- Varying business criticality
Security teams @ Adobe

ASSET
(Adobe Secure Software Engineering Team)

Researchers & PMs

Champions

Product Team

Engineering

Adobe Marketing Cloud

Adobe Document Cloud

Adobe Creative Cloud

Products
Establish the minimum bar

**SPLC Baseline Tasks for every team**

- Training
- Static analysis of code
- Security testing
- 3\(^{rd}\) party component tracking
- Code reviews
- Security requirements review
- Threat modelling
- Review of high risk findings and sign-off

- Create a SPLC standard that the product teams need to follow
- Standardize the tool chain
Security is a shared responsibility
Split and share responsibilities

<table>
<thead>
<tr>
<th>SPLC Tasks</th>
<th>Product team ownership</th>
<th>Central security team driven</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>Static analysis of code</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>Security testing</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>3rd party component tracking</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>Code reviews</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>Security requirements review</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Threat modelling</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Review of high risk findings and sign-off</td>
<td></td>
<td>✔</td>
</tr>
</tbody>
</table>

Spend premium security skill mindshare where it matters.
Set up product teams for security success with their security practices

Onboard Team

Review Product

- Gather intel
- Automation onboarding

Train team

- Routine SPLC tasks
The challenges for a security team

- Security team’s bandwidth
- Diverse technology
- Business criticality
Implementing Security Measures for a wide technology spectrum
A product may be offered on one or many platforms.

What is the product made up of?

- Mobile App
- Desktop Application
- Web / Service
Extend the baseline SPLC requirements

Baseline SPLC

Services SPLC Mobile SPLC Desktop SPLC
Extend the baseline SPLC requirements (web)
## Extend the baseline SPLC requirements (mobile)

<table>
<thead>
<tr>
<th>Sample Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Permissions</strong></td>
</tr>
<tr>
<td>Review all permissions requested by application.</td>
</tr>
<tr>
<td>☐ Reviewed?</td>
</tr>
<tr>
<td><strong>How secure is the current implementation / design?</strong></td>
</tr>
<tr>
<td>☐ Weak</td>
</tr>
<tr>
<td><strong>Security Notes</strong></td>
</tr>
</tbody>
</table>

| **Platform TLS requirements** |
| Review if app complies with recommended/mandated TLS requirements (ATS, network-config) |
| ☐ Reviewed? |
| **How secure is the current implementation / design?** |
| ☐ Weak | ☐ Fair | ☐ Strong |
| **Security Notes** |
Extend the baseline SPLC requirements (desktop)

<table>
<thead>
<tr>
<th>Signed Binaries</th>
<th>Secure Compiler Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Review how binaries are built and signed in the release process.</td>
<td>Review if product is leveraging secure compiler/linker flags on all platforms.</td>
</tr>
<tr>
<td>□ Reviewed?</td>
<td>□ Reviewed?</td>
</tr>
<tr>
<td><strong>How secure is the current implementation / design?</strong></td>
<td><strong>How secure is the current implementation / design?</strong></td>
</tr>
<tr>
<td>□ Weak □ Fair □ Strong</td>
<td>□ Weak □ Fair □ Strong</td>
</tr>
</tbody>
</table>

**Security Notes**

**Application Update**

Review how the application fetches and applies updates.

□ Reviewed?

**Local Web Server**

Review if application installs or embeds a local webserver.

□ Reviewed?
The challenges for a security team

- Security team’s bandwidth
- Diverse technology
- Business criticality
Tune for business criticality
Factor in business criticality for a security engagement

Dashboard products

New critical offerings

Products

- Baseline SPLC
- Services SPLC
- Mobile SPLC
- Desktop SPLC

Backlog creation

Backlog computation and tracking

Dashboard reporting

Deep threat models

Targeted third party pen tests

Acquisition

SPLC Onboarding

Pre-M&A security mitigations

Post-M&A security assessment
Summary

- We presented you with the real world experiences of running a SPLC program at Adobe.
- At a minimum, a product should get access to a baseline SPLC guidance.
- A SPLC program:
  - Scales premium security bandwidth through shared responsibility.
  - Evolves continuously as the company evolves and innovates.
  - Is flexible and adapts to the business needs of an organization.