AGILE AND CONTINUOUS THREAT MODELS

Nancy Davoust
Vice President, Security Architecture and Technology Solutions
Comcast
CONTEXT FOR AGILE AND CONTINUOUS THREAT MODELING
The Landscape is Chaotic

Exploding Number of Attack Surfaces and Attacks

Innovative but Insecure Technologies

Evolving Business Models

Agile & Continuous

Revolutionary Security Principles and Practices
Embracing Agile and Continuous Methodology

While we are Developing and Operating at the Speed of Light
Build Security In – Don’t Bolt It On

Education and Training

Pen Testing
Fuzzing
Code Review
Static / IAST Analysis

Compliance Validation
Recover
Respond
Detect
Monitor
Threat Intel

Policies
Log
Audit

Start

Threat Model
AGILE AND CONTINUOUS THREAT MODEL WORKSHOP
Threat Model Workshop In a Day with Each DevSecOps Team

1. Introduction, Goals and Background
2. Examples and Exercises
3. Live Threat Model
4. Risk Assessment
Threat Modeling Workshop Success Objectives

- Team trained to use agile and continuous threat modeling as a practice
- Reviewed architecture for real-world threats
- Common understanding of the threats and mitigations
- Protect customers and products earlier in the product lifecycle
- Team buy-in as the security findings were generated by the team
Everyone is Responsible for Security

- Open Posture
- Transparent
- One Team

- Be Honest
- No Blaming
- We are here to help one another

- Build security in by design
- Teamwork to identify attack surfaces
- We are all in this together
Threat Modeling Fundamentals

What is threat modeling?

- Defense in depth
- Mitigations
- Architecture & Features
- Use Cases
- Data
- Threats & Risk
- Attack Surfaces

Why do we need it?

- Reduce security design flaws
- Reduce cost to recover from attacks
- Create effective security requirements
- Know your enemies and their tactics

What is threat modeling?

- Use Cases
- Threats & Risk
- Attack Surfaces
- Mitigations
- Architecture & Features
- Data
- Defense in depth

Why do we need it?

- Reduce security design flaws
- Reduce cost to recover from attacks
- Create effective security requirements
- Know your enemies and their tactics
Security Breaches Can Happen Anywhere

- Banking
- Social Media
- Transportation
- Defense
- Education
- Food
- Retail
- Entertainment
- Services
- Technology
- Healthcare
- Email
- Manufacturing
- Utilities
- Retail
- Entertainment
- Services
- Technology
- Healthcare
- Email
- Manufacturing
- Utilities
- Social Media
- Transportation
- Defense
- Education
- Food
- Retail
- Entertainment
- Services
- Technology
- Healthcare
- Email
- Manufacturing
- Utilities
- Social Media
- Transportation
- Defense
- Education
- Food
- Retail
- Entertainment
- Services
- Technology
- Healthcare
- Email
- Manufacturing
- Utilities
# Common Weaknesses and Countermeasures

<table>
<thead>
<tr>
<th>Weaknesses</th>
<th>Countermeasures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insufficient API security</td>
<td>API security gateway, OAuth, Tokens, Certificates, Signing Keys</td>
</tr>
<tr>
<td>Exposed infrastructure &amp; admin ports</td>
<td>Jump boxes, network ACLs, security groups, iptables, MFA (deprecate telnet!)</td>
</tr>
<tr>
<td>Lack of privileged account management &amp; monitoring</td>
<td>Limit shared credentials, local accounts, monitor credential use for abuse. Forward logs to a centralized location, use correlation rules in a SIEM and defined alerts</td>
</tr>
<tr>
<td>Hard-coded credentials and API secrets</td>
<td>Key management solutions such as SafeNet, HashiCorp Vault, Ansible Vault, Puppet, Chef Data Bags, SALT, or your company recommended vault</td>
</tr>
<tr>
<td>Secure SDLC Practices not integrated into your CI/CD pipeline</td>
<td>Secure the pipeline (e.g. Jenkins, Ansible, Salt, GitHub, other tools), automate static code analysis, use scanning tools web app scanners, Nessus, Qualys)</td>
</tr>
</tbody>
</table>
## Attacker Profile Exercise

<table>
<thead>
<tr>
<th>Attacker</th>
<th>Attack Goals</th>
<th>Attacker Risk Tolerance</th>
<th>Attacker Level of Effort</th>
<th>Attacker Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyber Criminals</td>
<td>Financial</td>
<td>Low</td>
<td>Low → medium</td>
<td>Known proven</td>
</tr>
<tr>
<td>Industrial spies</td>
<td>Information &amp; Disruption</td>
<td>Low</td>
<td>High → extreme</td>
<td>Sophisticated &amp; unique</td>
</tr>
<tr>
<td>Hacktivists</td>
<td>Information, disruption, media attention</td>
<td>Medium → high</td>
<td>Low → medium</td>
<td>System administration errors and social engineering</td>
</tr>
<tr>
<td>Internal Attack/Insider</td>
<td>Information &amp; Disruption</td>
<td>High</td>
<td>High → extreme</td>
<td>Known proven</td>
</tr>
</tbody>
</table>
The Process

Threat Model Lead

Provides Guidance

Leads Discussion

Asks Questions

Identifies Vulnerabilities and Action Items

Assess Risk

Threat Model PM

Posts Architecture, Action Items and Findings and tracks issues to closure with the product team.
Threat Model Example
Identifying the Attack Surfaces

Service 1
Service 2
Service 3

Data Source 1
Data Source 2
Data Source 3

Middleware App

Data

HTTP
HTTPS
SSH
Local Logging
Update API
Admin Access
User API

1. HTTP (RISK)
2. HTTPS (RISK)
3. SSH (RISK)
4. Local Logging (RISK)
Attack Surface Exercise

1. HTTP
   - Unencrypted
   - Code Update Management
   - Self-signed TLS Certificates

2. Update API
   - Unauthorized Access
   - Stolen Data
   - Redirection Attacks

3. SSH
   - Root Access
   - Update Code
   - Configuration Changes

4. Logs
   - No Audit Trail
   - Unencrypted Sensitive Data
   - No Pruning of Data
Threat Impacts

- Personal Safety, Financial Safety
- Scaled Theft of Customer Data
- Scaled Denial of Service
- Scaled Theft of Service
- Malware
- Intellectual Property Theft
- Equipment Theft
Generalized Risk Equation
Risk = (Threat Impact * Likelihood) / Level of Effort
Summary

Today you learned about Threats, Impacts and Risk
How to Perform an Agile and Continuous Threat Model
Examples of attacks, vulnerabilities and effective countermeasures
Everyone is responsible for security

Build security in by design, don’t bolt it on