Secure SDLC Practices for the Cloud

Krishna Marella
September 18, 2014
We will discuss

• **An overview** of traditional security development practices

• **Challenges** in applying such practices into a cloud environment

• **Solutions and examples** of leading practices to consider for secure cloud adoption
Cyber risk is at the heart of software innovation and development trends

**Disruptive software trends**

<table>
<thead>
<tr>
<th>Software trends*</th>
<th>Cyber risk outcome examples:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-Channel Software Capability</td>
<td>Potentially ubiquitous software security and architecture vulnerabilities from non-standard development methods enabling external digital engagement capability</td>
</tr>
<tr>
<td>Increased Software Delivery Velocity</td>
<td>Iterative development methods (e.g., agile) require well-integrated security discipline for rapid sprints to build or update software functionality: analyze software security risk, test for security vulnerabilities and solution security-related defects</td>
</tr>
<tr>
<td>Increased Software Technical Debt</td>
<td>Poor software coding quality and architecture issues from the rush to innovate are often the root cause for software security weaknesses which expose cyber threat vulnerabilities</td>
</tr>
</tbody>
</table>

*Source: Deloitte Tech Trends 2014: Inspiring Disruption*
Traditional software assurance approach

Defined security gates and touch points that are focused on application layer during software development lifecycle are often only marginally effective.
The cloud challenge

- Rapidly increasing attack surface
- Privacy, compliance and data governance requirements
- New code for environment provisioning and management
- Resiliency is no longer an operational problem
- Tension between need for security and agility
- more…
Some recent examples

The following examples outline common design and implementation level issues identified with cloud applications.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Examples</th>
</tr>
</thead>
</table>
| 1. Weak identity and access controls | • Proliferation of user and service accounts  
• Weak authentication and credential management  
• No fine grained controls to cloud resources |
| 2. Cryptography and key management flaws | • Multi-purpose use and/or hardcoded keys  
• Inability to rotate or revoke keys  
• Insecure storage of key material |
| 3. Inadequate data protection controls | • Clear-text inter service communication  
• Sensitive data and secrets are not encrypted at rest  
• No tokenization or masking for data transfers |
| 4. Resiliency and continuity problems | • Lack of application level throttling  
• Untested RTO/RPO requirements  
• Single points of failure |
The following examples outline common design and implementation level issues identified with cloud applications.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>5. Multi-tenancy security risks</td>
<td>• Exposed APIs or service interfaces&lt;br&gt;• Caller authenticity could not be verified&lt;br&gt;• Trusting of shared domain or namespaces</td>
</tr>
<tr>
<td>6. Insecure SaaS application extensions</td>
<td>• New vulnerabilities in custom UI&lt;br&gt;• Insecure data exchange with on-premise systems</td>
</tr>
<tr>
<td>7. Infrastructure security issues</td>
<td>• Direct access from non-prod or on-premise systems&lt;br&gt;• Poor isolation or zoning of virtual infrastructure&lt;br&gt;• Migration of insecure configurations</td>
</tr>
<tr>
<td>8. Poor planning and due-diligence</td>
<td>• Security and compliance requirements are not vetted&lt;br&gt;• Unverified provider's security features&lt;br&gt;• Misunderstanding of shared responsibility model</td>
</tr>
</tbody>
</table>
Path for secure cloud adoption

1. Build foundation
2. Define security baselines
3. Enable security orchestration
4. Monitor and measure effectiveness
1. Build foundation

• Petition for a new secure development model
  – Collaborate with technology & risk leaders
  – Think beyond “bolting old security tactics”

• Enhance secure SDLC practices for cloud
  – Address broader risk areas

- Software Security
- Operational Security
- Resiliency & Availability
- Monitoring & Licensing
- IP Protection
- Trust, Safety and Compliance
1. Build foundation (cont.)

- Build new skills for securing cloud applications
  - Cloud training and education
  - Engage cloud specialists
- Customize practices based on risk profile

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Operating Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk &amp; Compliance</td>
<td></td>
</tr>
<tr>
<td>- Sensitive data</td>
<td>- Development methodology</td>
</tr>
<tr>
<td>- Compliance obligations</td>
<td>- Deployment model</td>
</tr>
<tr>
<td>- Privacy</td>
<td>- Sourcing model</td>
</tr>
<tr>
<td>- Safety &amp; abuse protection</td>
<td>- Extent of customization</td>
</tr>
<tr>
<td>- Global trade compliance</td>
<td>- Release frequency</td>
</tr>
<tr>
<td>Technology</td>
<td></td>
</tr>
<tr>
<td>- Cloud service</td>
<td>- Anticipated Use</td>
</tr>
<tr>
<td>- Development platform</td>
<td>- Accessibility</td>
</tr>
<tr>
<td>- Client-side libraries</td>
<td>- Availability commitments</td>
</tr>
<tr>
<td>- Integration with other systems</td>
<td>- Geography</td>
</tr>
<tr>
<td>- Software licensing</td>
<td>- User types</td>
</tr>
<tr>
<td>- Infrastructure</td>
<td>- Number of users</td>
</tr>
</tbody>
</table>
2. Define security baselines

- Enhance security policies and standards for cloud
- Identify and harden virtual assets
- Develop consistent risk assessment approaches
- Seek guidance from industry resources

Illustrative Example

- Cloud Security Alliance (CSA) Cloud Control Matrix (CCM)
- NIST SP 800-53 and SP 800-144
- ISO/ IEC 27017 (expected in 2015)
2. Define security baselines (cont.)

- Develop specific design patterns for secure cloud integration

Illustrative Examples

- Just-in-time provisioning
- Federated SSO
- Token based authorization
- Persona vs. identity
- Application telemetry
3. Enable security orchestration

• Mandate security controls for privileged activities, such as:
  – Strong passwords
  – MFA
  – Adaptive MFA
  – Activity tracking

• Automate security tasks through smart integration, such as:
  – IDE plugins
  – Build
  – Deployment
3. Enable security orchestration (cont.)

Illustrative Example

Automation of security configuration through a Chef Cookbook to perform security patch updates, web server hardening and WAF install

```
bash "modify_apache_configuration" do
  user "root"
  code <<=EOH
    if grep -q "ServerTokens" /etc/httpd/conf/httpd.conf; then
      sed -c -i "s/(ServerTokens *)/*/1 PROD/" /etc/httpd/conf/httpd.conf
    else
      echo "ServerTokens PROD" >>=/etc/httpd/conf/httpd.conf
    fi

    if grep -q "ServerSignature" /etc/httpd/conf/httpd.conf; then
      sed -c -i "s/(ServerSignature *)/*/1 OFF/" /etc/httpd/conf/httpd.conf
    else
      echo "ServerSignature Off" >>=/etc/httpd/conf/httpd.conf
    fi
  EOH
end

bash "mod_security" do
  user "root"
  cwd "/tmp"
  code <<=EOH
    yum -y install mod_security
    service httpd restart
  EOH
end
```
4. Monitor and measure effectiveness

**Provider Transparency**
- SLAs and Service Commitments
- Security Whitepapers
- Product Security Features
- Industry Certifications
- Third-party Reports

**Security & Continuity Testing**
- Network scanning
- Vulnerability assessment
- Penetration testing
- Disaster recovery testing

**Metrics & Alerting**
- Security Policy Compliance
- Configuration
- User Activity
- Anomaly Detection
- Analytics
- Application Diagnostics
Takeaways

• Incorporate broader risk considerations
• Make security practices consumable for developers
• Collaboration is needed for effective security
• Cloud can also lead to new security opportunities
Questions?
This presentation contains general information only and Deloitte is not, by means of this presentation, rendering accounting, business, financial, investment, legal, tax, or other professional advice or services. This presentation is not a substitute for such professional advice or services, nor should it be used as a basis for any decision or action that may affect your business. Before making any decision or taking any action that may affect your business, you should consult a qualified professional advisor.

Deloitte shall not be responsible for any loss sustained by any person who relies on this presentation.

About Deloitte
Deloitte refers to one or more of Deloitte Touche Tohmatsu Limited, a UK private company limited by guarantee (“DTTL”), its network of member firms, and their related entities. DTTL and each of its member firms are legally separate and independent entities. DTTL (also referred to as “Deloitte Global”) does not provide services to clients. Please see www.deloitte.com/about for a detailed description of DTTL and its member firms. Please see www.deloitte.com/us/about for a detailed description of the legal structure of Deloitte LLP and its subsidiaries. Certain services may not be available to attest clients under the rules and regulations of public accounting.

Copyright © 2014 Deloitte Development LLC. All rights reserved.
Member of Deloitte Touche Tohmatsu Limited