Building and operating cloud apps securely using the Secure DevOps Kit for Azure

Jonathan Trull
“Companies looking to digitally transform need a trusted cloud” – Satya Nadella
<table>
<thead>
<tr>
<th>Onboarding</th>
<th>Configuration</th>
<th>Operational Practices and Hygiene</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pilot Onboarding</td>
<td>Get Secure Process</td>
<td>Incident Response Table Top</td>
</tr>
<tr>
<td>Subscription Procurement</td>
<td>Subscription Configuration</td>
<td>Backup and Recovery Table Top</td>
</tr>
<tr>
<td>Security Operations Onboarding</td>
<td>Azure Security Center Configuration</td>
<td>Audit Log Table Top</td>
</tr>
<tr>
<td>Configuration Insights Onboarding</td>
<td>Application Insights Configuration</td>
<td>Azure Security Center Table Top</td>
</tr>
<tr>
<td>OMS Onboarding</td>
<td>Deployment Template Use</td>
<td>Configuration Insights Table Top</td>
</tr>
<tr>
<td>MyApps Onboarding</td>
<td>PowerShell Library Use</td>
<td>Subscription User Access Reviews</td>
</tr>
<tr>
<td>Engage ISRM Onboarding</td>
<td>Network Configuration</td>
<td>RG/Resource Access Reviews</td>
</tr>
<tr>
<td>SAW Onboarding</td>
<td>Forensics Use</td>
<td>Key and Secret Rotation Table Top</td>
</tr>
<tr>
<td>Alternate Credentials – SC-backed</td>
<td>Key and Secret Management</td>
<td>Release Management/Change Control</td>
</tr>
<tr>
<td>ExpressRoute Onboarding</td>
<td>Key Escrow Process</td>
<td>Segregation of Duties &amp; Environments</td>
</tr>
<tr>
<td>Identification of user roles</td>
<td>Tech Control Procedures and Validation</td>
<td>OSA Control Procedures</td>
</tr>
<tr>
<td></td>
<td>ARM Policy and Resource Lock</td>
<td>Resource Utilization - MyBill</td>
</tr>
<tr>
<td></td>
<td>Resource Group Configuration</td>
<td></td>
</tr>
<tr>
<td></td>
<td>OMS Configuration</td>
<td></td>
</tr>
</tbody>
</table>
Learnings from pilot projects

> **Automating security** is a must!
  - Challenging to maintain parity across “DevTest -> SIT -> UAT -> PROD”. Too many environments to review!
  - Need to strive for end-to-end validation – from a claim during a threat model to actual resource in the subscription
  - HBI v/s other resources in a sub – tagging/organization can help. Need to track critical HBI artifacts (secure ‘snapshot’)
  - Exception processes such as DRA causes lot of angst

> Need to push **engineering team empowerment** to the next level
  - Make pre-configuration of security easier – e.g., miniature ARM-templates
  - Integrate security tools such as CredScan into OneITVSO build workflow
  - Expand the scope of automated security control verification (more services, more controls for each service)

> Continuous change requires a shift towards **continuous assurance**!
  - “Signed-off” designs and “point-in-time” assessments have a natural tension with CICD
  - Need a capability to capture security snapshots and track ‘drift’ from a secure state

> Need to understand and establish **operational security hygiene** in Azure
  - Subscription hygiene, access reviews, change management
  - Resource configurations, firewalls, network (appliance) configuration reviews
  - Key rotation, BC/DR, auditing/monitoring, incident response
Secure DevOps Kit for Azure

1. **Subscription Security** (Policy, ASC Config, Alerts, RBAC, etc.)
   - Provision security in subscription

2. **Security IntelliSense, Security Verification Tests (SVTs)**
   - Develop securely, spot check security via scripts

3. **CICD Build/Release Extensions**
   - Deploy securely from VSO build/release pipeline

4. **Continuous Assurance Runbooks**
   - Periodically scan in production to watch for drift

5. **OMS Solution for Alerting & Monitoring**
   - Single security dashboard across DevOps stages

6. **Cloud Risk Governance**
   - Make data-driven improvements to security

{JSON}
Secure DevOps Kit for Azure

1. Provision security in subscription
2. Develop securely, spot check security via scripts
3. Deploy securely from VSO build/release pipeline
4. Periodically scan in production to watch for drift
5. Single security dashboard across DevOps stages
6. Make data-driven improvements to security

- Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)
- Security IntelliSense, Security Verification Tests (SVTs)
- Continuous Assurance Runbooks
- Cloud Risk Governance
- OMS Solution for Alerting & Monitoring
- CICD Build/Release Extensions

Secure DevOps Kit for Azure
1- Secure your cloud subscription

RBAC  ARM Policy  Resource Locks  Contact Phone  Alerts
Subscription security setup script...

```powershell
Set-AzSKSubscriptionSecurity -SubscriptionId <sub_id> ...
```

```
PS C:\> Set-AzSDKSubscriptionSecurity -SubscriptionId $subId -SecurityContactEmails 'mprabhu@microsoft.com, sbyna@microsoft.com' -SecurityPhoneNumber '425-863-1148'

AzSDK Version: 2.8.1
Method Name: Set-AzSDKSubscriptionSecurity

Running AzSDK cmdlet using CSE policy...

Configuring Security Center

SecurityCenterPolicy configuration in your subscription is already up to date. If you would like to reconfigure, please rerun the command with '-Force' parameter.

Completed Security Center configuration

Setting up subscription RBAC

Processing RBAC rules for adding central accounts. Tags: [Mandatory]. Total accounts: 1
All required accounts are correctly configured

Completed subscription RBAC configuration

Setting up ARM policies

Processing AzSDK ARM policies. Total policies: 6
Note: Configuring ARM policies can take about 2-3 min...
32% Completed
67% Completed
100% Completed
All AzSDK ARM policies have been added to the subscription successfully

Completed ARM policy configuration

Setting up Alerts

Processing AzSDK alerts. Total alerts: 7
Note: Configuring alerts can take about 4-5 min...
All AzSDK alerts have been configured successfully.

Completed Alerts configuration
```
Secure DevOps Kit for Azure

1. Provision security in subscription
2. Develop securely, spot check security via scripts
3. Deploy securely from VSO build/release pipeline
4. Periodically scan in production to watch for drift
5. Single security dashboard across DevOps stages
6. Make data-driven improvements to security
2- Empower developers – code, compile, prototype

Along with the usual suspects...

- Static code analysis tools
- Traditional security checks
- Credentials in code, etc.

public static void RandomData() {
    // Secure
    var random = new RNGCryptoServiceProvider();
    // Secure
    var rng = new RNGCryptoServiceProvider();
    // Secure
    var sha256 = new SHA256Managed();
    // Secure
    var sha = new SHA1Managed();
    // Secure
    var aes = new AesManaged();
    // Secure
    var key = new Key;
    // Secure
    var config = new AesConfig

    // Secure
    var rs = new RSACryptoServiceProvider();
    // Secure
    var rs2 = new RSACryptoServiceProvider();
public static void RandomData()
{
    // Insecure Random data generator
    var random = new Random();

    // Secure
    var rng = new RNGCryptoServiceProvider();

    // Insecure hashing algo
    var md5 = new MD5CryptoServiceProvider();

    // Insecure hashing algo
    var sha1 = new SHA1CryptoServiceProvider();

    // Secure
    var sha256 = new SHA256CryptoServiceProvider();

    // Insecure Encryption
    var rijndael = new RijndaelManaged();

    // Secure
    var aes = new AesCryptoServiceProvider();

    // Insecure AES config
    aes.KeySize = 128;

    // Suggested fix
    var rng = new RNGCryptoServiceProvider();
    var md5 = new MD5CryptoServiceProvider();
    var sha256 = new SHA256CryptoServiceProvider();
    var aes = new AesCryptoServiceProvider();
    aes.KeySize = 128;
Get-AzSKAzureServicesSecurityStatus -SubscriptionId <sub_id> -ResourceGroupNames <rg1, rg2>

Security test coverage for 25+ Azure services
Outputs generated

CSV file

Detailed LOG

```
"DisplayName" = "AzSDKCICDApp";
"SignInName" = $Null;
"ObjectType" = "ServicePrincipal"
}

"DisplayName" = "AzSDKCCTest_eKzX0/7B8zTeGoUdA7yfY+Lsht4Q8LVSSNAW/7hgg8=";
"SignInName" = $Null;
"ObjectType" = "ServicePrincipal"
}

**Failed**: [SubscriptionCore]: [Service Principal Names (SPNs) should not be Owners/Contributors on the subscription] for subscription: [MSFT - SECURITY REFERENCE ARCHITECTURE - 02]

---

[Azure Subscription_SI_Lock_Critical_Resources]: Critical application resources should be protected using a resource lock

05/22/2017 11:41:05-Info: Checking: [SubscriptionCore]-[Critical application resources]```
Secure DevOps Kit for Azure (AzSDK)

Security Report

Subscription Name: MRS0K-
SubscriptionId: 6fca40b-1ba0-81f4-54a4-c8e4af5-7c5f4ad
generated by:
Azure
Generated on:
July 31, 2017 17:45 (UTC)
Requested by:
mpinho121@live.com (User)
Command Executed:
Get-AzSDKAzureServicesSecurityStatus
-SubscriptionId '6fca40b-1ba0-81f4-54a4-c8e4af5-7c5f4ad'
-GeneratedPDF Portrait
Documentation:
http://aka.ms/azsdkdocs
FAQ:
http://aka.ms/azsdkdocs/faq
Support DL:
mailto:mmazzdkosp@microsoft.com

# AzSDK repair function uses files from adjacent 'services' folder
# Repair Azure resources
Repair-AzSDKAzureServicesSecurity
-ParameterFilePath  '"SScriptRoot\FixControlConfig.json"' #
-ResourceTypeNames "" 
-ResourceGroupNames ""
-ResourceNames ""
-ControlIds ""

RunFixScript.ps1
Secure DevOps Kit for Azure

1. Provision security in subscription

2. Develop securely, spot check security via scripts

3. Deploy securely from VSO build/release pipeline

4. Periodically scan in production to watch for drift

5. Single security dashboard across DevOps stages

6. Make data-driven improvements to security

- Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)
- Security IntelliSense, Security Verification Tests (SVTs)
- OMS Solution for Alerting & Monitoring
- Continuous Assurance Runbooks
- Cloud Risk Governance
- CICD Build/Release Extensions

OMS Solution for Alerting & Monitoring

Make data-driven improvements to security

Single security dashboard across DevOps stages

Periodically scan in production to watch for drift

Develop securely, spot check security via scripts

Deploy securely from VSO build/release pipeline

Provision security in subscription
3 - Bake security into cloud deployments (CICD)

- Code Complete
- Dev Envmt
- Test Envmt
- Prod Envmt
Validate deployment templates
Fix deployment templates

```
PS D:\repos\BuildWebAppDemo\Build\BuildDemoApp\BuildAppARMTemplates> Get-AzSKARMTemplateSecurityStatus -ARMTemplatePath '.\azuredeploy_compliant.json' -Preview
AzuSK Version: 3.1.0

Method Name: Get-AzSKARMTemplateSecurityStatus
Input Parameters:
Key          Value
---          -----
ARMTemplatePath '.\azuredeploy_compliant.json'
Preview       True

Starting analysis: [Filename: D:\repos\BuildWebAppDemo\Build\BuildDemoApp\BuildAppARMTemplates\azuredeploy_compliant.json]

Passed: [Azure_Storage_DP_Encrypt_At_Rest_Blob]
Passed: [Azure_Storage_DP_Encrypt_In_Transit]
Passed: [Azure_AppService_BCDR_Use_Multiple_Instances]
Passed: [Azure_AppService_Config_Enable_Devirtualization]
Passed: [Azure_AppService_Config_Enable_Remote_Debugging]
Passed: [Azure_AppService_Config_Enable_Devirtualization]
Passed: [Azure_AppService_BCDR_Use_Awareness]
Passed: [Azure_AppService_BCDR_Use_Latest_Version]
Passed: [Azure_AppService_Audit_Enable_Locking_and_Monitoring]
Passed: [Azure_AppService_Audit_Enable_Locking_and_Monitoring]
Passed: [Azure_AppService_Audit_Enable_Locking_and_Monitoring]
Passed: [Azure_AppService_DP_Dont_Allow_HTTP_Access]
Passed: [Azure_AppService_AuthN_Use_AAD_for_Client_AuthN]

Summary Total Passed
High      6     6
Medium    7     7
Low       1     1
Total     14    14
```

"Secure" Deploy

Create ARM Templates
Secure ARM Templates
Deploy using CICD
Deploy through CICD pipeline (non-compliant)

---

### BuildDemoWebAppNonCompliant / Release-2

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Dev</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fire-deployment approval</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Agent phase</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Initialize Agent</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Initialize Job</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Download artifact - BuildWebAppDe..</td>
<td></td>
</tr>
<tr>
<td></td>
<td>A2SK_ARMTemplateChecker</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Azure Deployment:Create Or Update ...</td>
<td></td>
</tr>
<tr>
<td></td>
<td>A2SK_SVTs</td>
<td></td>
</tr>
</tbody>
</table>

---

Agent queue: Hosted V2017 | Agent: Hosted Agent

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>154</td>
<td>2016-04-17T01:18:32.4204812 Task</td>
<td>A2SK ARM Template Checker</td>
</tr>
<tr>
<td>155</td>
<td>2016-04-17T01:18:32.4207602 Description</td>
<td>Scan ARM templates for security issues using A2SK.</td>
</tr>
<tr>
<td>156</td>
<td>2016-04-17T01:18:32.4208592 Version</td>
<td>1.4.1.1</td>
</tr>
<tr>
<td>157</td>
<td>2016-04-17T01:18:32.42040852 Author</td>
<td>Microsoft Corporation</td>
</tr>
<tr>
<td>159</td>
<td>2016-04-17T01:18:32.4204789 Installing Module A2SK</td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>2016-04-17T01:18:32.4207737 Installing Module A2SK</td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>2016-04-17T01:18:32.42080952 Installing Module A2SK</td>
<td></td>
</tr>
<tr>
<td>162</td>
<td>2016-04-17T01:18:32.4209714 A2SK Version</td>
<td>3.1.0</td>
</tr>
<tr>
<td>163</td>
<td>2016-04-17T01:18:32.4209542 A2SK Version</td>
<td>3.1.0</td>
</tr>
<tr>
<td>164</td>
<td>2016-04-17T01:18:32.4209938 Starting analysis: [FileName: azuredeploy_noncompliant.json]</td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>2016-04-17T01:18:32.4210662 Passed</td>
<td>[Azure Storage DP_Encrypt At Rest]</td>
</tr>
<tr>
<td>166</td>
<td>2016-04-17T01:18:32.4210662 Failed</td>
<td>[Azure Storage DP_Encrypt At Rest]</td>
</tr>
<tr>
<td>167</td>
<td>2016-04-17T01:18:32.4210662 Passed</td>
<td>[Azure Storage DP_Encrypt At Rest]</td>
</tr>
</tbody>
</table>

---

"Secure" Deploy

Create ARM Templates
Secure ARM Templates
Deploy using CICD
Deploy through CICD pipeline (compliant)

Create ARM Templates
Secure ARM Templates
Deploy using CICD

“Secure” Deploy
VSTS release task for ‘AzSK SVTs’

Security Verification Tests (SVTs)
Release promoted (or not) based on outcome of AzSK SVT scan.
Secure DevOps Kit for Azure

1. Provision security in subscription

2. Develop securely, spot check security via scripts

3. Deploy securely from VSO build/release pipeline

4. Periodically scan in production to watch for drift

5. Single security dashboard across DevOps stages

6. Make data-driven improvements to security

- Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)
- Security IntelliSense, Security Verification Tests (SVTs)
- CICD Build/Release Extensions
- OMS Solution for Alerting & Monitoring
- Continuous Assurance Runbooks
- Cloud Risk Governance

OMS Solution for Alerting & Monitoring

Cloud Risk Governance

Secure DevOps Kit for Azure

OMS Solution for Alerting & Monitoring
4- Setup continuous security coverage

Azure Automation

Scan cloud resources in a scheduled fashion
Continuous assurance setup (a peek)

Features:
- Single-click, self-managing setup
- Scan subscription & resources (RGs)
- Auto-update of AzureRm & AzSK modules
- OMS integration
- Detailed reports in Storage Blobs

```powershell
# Continuous Assurance setup
$somsWSId = '7b797' .................
$somsShrKey = '40PXUilhyMyPp' .................

Install-AzSKContinuousAssurance
-ResourceGroupNames $appRGS #RG1, RG2 or '*
-SubscriptionId $orgSub1
-OMSWorkspaceId $somsWSId
-OMSSharedKey $somsShrKey
```
4- Continuous assurance setup (a peek)

**Org baseline mode (e.g.)**

- Azure_Subscription_AuthZ_Add_Required_Central_Accounts
- Azure_Subscription_AuthZ_Remove_Deprecated_Accounts
- Azure_Subscription_AuthZ_Dont_Use_NonAD_Ids
- Azure_Subscription_AuthZ_Remove_Management_Certs
- Azure_Subscription_Config_Azure_Security_Center
- Azure_Subscription_Config_ARM_Policy
- Azure_Subscription_Audit_Configure_Critical_Alerts

- Azure_VirtualMachine_Deploy_Latest_OS_Version
- Azure_VirtualMachine_Config_OS_Auto_Update_Windows
- Azure_VirtualMachine_Config_Enable_Antimalware_Windows
- Azure_VirtualMachine_NetSec_Dont_Open_Management_Ports

- Azure_SQLDatabase_DP_Enable_TDE
- Azure_SQLDatabase_Audit_Enable_Threat_Detection_Server
- Azure_SQLDatabase_Audit_Enable_Threat_Detection_DB

- Azure_AppService_DP_Dont_Allow_HTTP_Access
- Azure_Storage_AuthN_Dont_Allow_Anonymous
- Azure_Storage_DP_Encrypt_At_Rest_Blob
- Azure_Storage_DP_Encrypt_At_Rest_File

- Azure_CloudService_DP_DontAllow_HTTP_Access_InputEndpoints
- Azure_CloudService_SI_Auto_OSUpdate
- Azure_CloudService_SI_Enable_AntiMalware
- Azure_CloudService_SI_Disable_RedeemRemoteDesktop_Access
CA – standalone v. central-scan modes

AzSK CA

standalone v. central-scan modes

AzSK CA
Control Attestation (overriding AzSK results)

<table>
<thead>
<tr>
<th>AzSK Ctrl-Result</th>
<th>Sub-status</th>
<th>Effective Ctrl-Result</th>
<th>Color</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passed</td>
<td>None</td>
<td>Passed</td>
<td>G</td>
<td></td>
</tr>
<tr>
<td>Verify</td>
<td>None</td>
<td>Verify</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>Verify</td>
<td>NotAnIssue</td>
<td>Passed</td>
<td>G</td>
<td>Justification</td>
</tr>
<tr>
<td>Verify</td>
<td>NotFixed</td>
<td>Exception</td>
<td>Y</td>
<td>Justification</td>
</tr>
<tr>
<td>Failed</td>
<td>None</td>
<td>Failed</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>Failed</td>
<td>NotAnIssue</td>
<td>Passed</td>
<td>G</td>
<td>Justification</td>
</tr>
<tr>
<td>Failed</td>
<td>NotFixed</td>
<td>Exception</td>
<td>Y</td>
<td>Justification</td>
</tr>
<tr>
<td>Error</td>
<td>None</td>
<td>Error</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>NotAnIssue</td>
<td>Passed</td>
<td>G</td>
<td>Justification</td>
</tr>
<tr>
<td>Error</td>
<td>NotFixed</td>
<td>Exception</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Manual</td>
<td>None</td>
<td>Manual</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>Manual</td>
<td>NotAnIssue</td>
<td>Passed</td>
<td>G</td>
<td>Justification</td>
</tr>
<tr>
<td>Manual</td>
<td>NotFixed</td>
<td>Exception</td>
<td>Y</td>
<td></td>
</tr>
</tbody>
</table>

Technical Result

\{Passed, Verify, Failed, Manual, Error\}

Human Input

\{None, NotAnIssue, Exception\}

Final Result

\{Passed, Verify, Failed, Manual, Error, RiskAck\}
## Control Attestation – scenarios, workflow

<table>
<thead>
<tr>
<th>Control (e.g.,)</th>
<th>Technical Evaluation Result</th>
<th>User action/choice</th>
<th>Effective Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBAC grants</td>
<td>‘Verify’</td>
<td>Reviewed log output, looks ok</td>
<td>‘Passed’</td>
</tr>
<tr>
<td>SSE on Storage</td>
<td>‘Failed’</td>
<td>Override based on contextual info</td>
<td>‘Passed’</td>
</tr>
<tr>
<td>Use of WebSocket</td>
<td>‘Failed’</td>
<td>Cannot fix at present, filing for risk ack</td>
<td>‘Exception’</td>
</tr>
</tbody>
</table>

### Features:
- State-based attestation
- Tracking justifications
- Auto-expiry of attestations
- Access control
- Auditing of attestation state changes

### Starting Control Attestation workflow...

Note: Enter 9 during the attestation workflow to abort.

Info: Starting attestation - [FeatureName: SubscriptionCore] [SubscriptionName:]

- **No. of controls that need to be attested:** 6
- **ControlId:** Azure Subscription Allow Limit Admin Owner Count
- **ControlSeverity:** Medium
- **Description:** Minimize the number of admins/owners
- **CurrentControlStatus:** Failed

Please select an action from below:
1. None
2. Attest
User Choice: 1

Please select an attestation status from below:
1. None
2. NotApplicable
3. NotIssued
4. Notified
User Choice: 2

Justification: This would impact our application release timelines. As discussed with security team, adding it to the backlog for next sprint.
Secure DevOps Kit for Azure

1. Provision security in subscription
2. Develop securely, spot check security via scripts
3. Deploy securely from VSO build/release pipeline
4. Periodically scan in production to watch for drift
5. Single security dashboard across DevOps stages
6. Make data-driven improvements to security

- Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)
- Security IntelliSense, Security Verification Tests (SVTs)
- CICD Build/Release Extensions
- Continuous Assurance Runbooks
- OMS Solution for Alerting & Monitoring
- Cloud Risk Governance
- Periodically scan in production to watch for drift
- Make data-driven improvements to security
- Single security dashboard across DevOps stages
5- Monitor security across dev ops stages

- Individual developer
- Security automation in CICD
- Continuous Assurance

Operations Management Suite (OMS) Repository

* x 1..N apps
Alerts from OMS (DevOps Kit control failures)

Wed 6/14/2017 8:07 PM

Microsoft Operations Management Suite Team <noreply@oms.microsoft.com>

To: Subhendranath Byna

If there are problems with how this message is displayed, click here to view it in a web browser.

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>Top 10 result(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SourceSystem: RestAPI</td>
<td></td>
</tr>
<tr>
<td>TimeGenerated: 6/14/2017 2:31:14 PM</td>
<td></td>
</tr>
<tr>
<td>ActualVerificationResult: Failed</td>
<td></td>
</tr>
<tr>
<td>ControlSeverity: High</td>
<td></td>
</tr>
<tr>
<td>ResourceType: Microsoft.Storage/storageAccounts</td>
<td></td>
</tr>
<tr>
<td>ResourceGroup: AsSDK-Demo-RG</td>
<td></td>
</tr>
<tr>
<td>Reference: aka.ms/azsdkossttcp/storage</td>
<td></td>
</tr>
<tr>
<td>ResourceName: azsdkdemosaft7ail5se</td>
<td></td>
</tr>
<tr>
<td>ControlStatus: Failed</td>
<td></td>
</tr>
<tr>
<td>ControlId: Azure_Storage_DP_Encrypt_At_Rest_Blob</td>
<td></td>
</tr>
<tr>
<td>SubscriptionName: MSFT-Security Reference Architecture-04</td>
<td></td>
</tr>
<tr>
<td>FeatureName: Storage</td>
<td></td>
</tr>
<tr>
<td>Source: CC</td>
<td></td>
</tr>
<tr>
<td>Recommendation: Run command 'Set-AzureRmStorageAccount -Name '&lt;StorageAccountName&gt;' -ResourceGroupName '&lt;RGName&gt;' -EnableEncryptionService 'isleb'. Run 'Get-Help Set-AzureRmStorageAccount -full' for more help.</td>
<td></td>
</tr>
<tr>
<td>SubscriptionId: 254ad434-e2e6-45c0-a32b-34bf24cb7479</td>
<td></td>
</tr>
<tr>
<td>Id: b4b5019e-874e-d316-3e81-8beac3ae8c32</td>
<td></td>
</tr>
<tr>
<td>Type: AsSDK_CL</td>
<td></td>
</tr>
<tr>
<td>MG: 00000000-0000-0000-0000-000000000000</td>
<td></td>
</tr>
</tbody>
</table>
Secure DevOps Kit for Azure

1. Provision security in subscription
2. Develop securely, spot check security via scripts
3. Deploy securely from VSO build/release pipeline
4. Periodically scan in production to watch for drift
5. Single security dashboard across DevOps stages
6. Make data-driven improvements to security

- Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)
- Security IntelliSense, Security Verification Tests (SVTs)
- CICD Build/Release Extensions
- Continuous Assurance Runbooks
- OMS Solution for Alerting & Monitoring
- Cloud Risk Governance

*OMS Solution for Alerting & Monitoring makes data-driven improvements to security*

*Single security dashboard across DevOps stages*

*Develop securely, spot check security via scripts*

*Deploy securely from VSO build/release pipeline*

*Periodically scan in production to watch for drift*

*Make data-driven improvements to security*
NOTE: This SS status tab represents whether the security configurations of your subscriptions meet the current security baseline being driven for CSE. See the Resource Security tab to view the security state of resources in the subscription.

IsSubCompliant represents the cumulative status for the subscription controls applicable to the selected drive (Wave 1/Wave 2/etc.). Individual status columns for Alerts, ARM, ASC, etc., will be removed soon. To determine the individual controls that need fixing, please utilize your OMS workspace.
Cloud risk governance: 'baseline compliance' status

NOTE: This RS status tab represents whether the security configurations of your resources meet the current security baseline being driven for CSE.

Cumulative numbers by resource type are shown here. To determine the individual controls that need fixing, please utilize your OMS workspace.
Being 'data driven' about security initiatives

Prevalent areas of "cloud risk"
Secure DevOps Kit for Azure

Subscription Security
(Policy, ASC Config, Alerts, RBAC, etc.)

1. Provision security in subscription

Cloud Risk Governance

6. Make data-driven improvements to security

OMS Solution for Alerting & Monitoring

5. Single security dashboard across DevOps stages

Security IntelliSense, Security Verification Tests (SVTs)

2. Develop securely, spot check security via scripts

CICD Build/Release Extensions

3. Deploy securely from VSO build/release pipeline

Continuous Assurance Runbooks

4. Periodically scan in production to watch for drift

OMS Solution for Alerting & Monitoring

6. Make data-driven improvements to security

Single security dashboard across DevOps stages
Secure DevOps Kit – Impact at Microsoft IT

1000+ subscriptions scanned

35000 Azure resources secured

50+ enterprise LOB app SDLs across IT

25 million+ controls scanned till date

200k+ hours of manual effort saved

250+ security controls across 30 Azure PaaS/IaaS service types
Coverage of Azure Services in the DevOps Kit
Learnings from our DevOps Kit experience

Predominant motivation is *still* to seek ‘Security sign-off’

Need to invest more in awareness about ownership of risk

Having ‘Dev’ and ‘Ops’ together is *not* ‘Dev Ops’

Breaking down ‘classic’ team silos is going to be critical

Understanding of cloud security model is limited

Need more education/trainings in this area

Systemic gap in PowerShell/scripting expertise

Need to invest in scripting skills (DevOps == “Infrastructure as Code”)

Engage uniformly with all stakeholders

We started with a ‘dev heavy’ approach, should have taken ops along
Next steps...

Try out the Secure DevOps Kit for Azure!

- GitHub src/docs: https://github.com/azsk
- Controls coverage: https://aka.ms/devopskit/tcp
- IT Showcase: https://aka.ms/devopskit/itshowcase
- Support: azsksup@microsoft.com
Thanks for attending!
References
Core use cases – local scan

DevOps Kit PS Module (+ AzureRm)

Your Subscription

Attestations

DSRE Subscription (CSE policy, dashboard telemetry)
Core use cases – Continuous Assurance

DevOps Kit PowerShell Module (+ AzureRm)
Core use cases – CICD Extension

- **PowerShell Gallery**
- **Build Agent**
- **Your Sub**
- **Attestations**
- **OMS**
- **DSRE Subscription** (CSE policy, dashboard telemetry)

Diagram showing integration with various tools and platforms.
DevOps Kit OMS Solution

Microsoft Operations Management Suite

About the Azure Security Monitoring View

Security Monitoring using the AzSDK

Express Route Network Security

Resource Security (V1.1)

Subscription Security (V1.0)

How to use this view...

This view displays various metrics related to Azure subscription security compliance. It provides a comprehensive overview of security risks and compliance status.

Key Controls:
- Network Security
- Azure AD Security
- Identity and Access Management

Interactive Features:
- View Details
- Download Report

Security Monitoring using the AzSDK

Express Route Network Security

Resource Security (V1.1)
AzSK command acronyms
Customizing & extending AzSK
Secure DevOps Kit for Azure (AzSK)

A toolkit for accelerating adoption of Azure at the enterprise by automating cloud resource configuration security for dev ops environments.

Areas of Focus
✓ Subscription security (ARM policies, RBAC, Alerts, ASC setup)
✓ Secure development (security baseline scans for ~30 Azure services)
✓ Security in deployment (AzSK CICD task for VSO)
✓ Continuous assurance in Ops (Azure Automation runbooks)
✓ Alerting & monitoring (single pane view in OMS across dev ops stages)
✓ Cloud risk governance (dashboards based on exhaustive security telemetry)

Key Benefits
▪ Accelerates cloud dev ops for the enterprise
▪ Empowers engineering teams to perform consistent cloud security configuration across dev ops stages
▪ Scale out security expertise through automation
▪ Facilitates risk governance in the cloud through security telemetry

Partners & Collaboration
Built with contributions from multiple teams:
▪ DSRE: GRCC, ACE, SAFET, Tools, ISO, etc.
▪ CSE: CloudMS, NIS, E3, App Teams (EDDA, CELA BI, Incentive Comp, etc.)
▪ PG: ASC, OMS, Visual Studio, Key Vault, Azure Automation, etc.
▪ Other: Account Teams, CAT, GD, etc.

Showcase to Customers/Conferences
▪ Showcased to key enterprise customers (GEICO, Symantec, TCS, Infosys, DELL, Shell)
▪ Used by Global Delivery (GD) for their envt
▪ Presented at Inspire, RSA and Ready conferences, internal engineering forums.
▪ IT Showcase: Secure DevOps for Azure
▪ Built end to end on top of Microsoft Azure
Secure DevOps Kit for Azure

1. **Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)**
   - Provision security in subscription

2. **Security IntelliSense, Security Verification Tests (SVTs)**
   - Develop securely, spot check security via scripts

3. **CICD Build/Release Extensions**
   - Deploy securely from VSO build/release pipeline

4. **Continuous Assurance Runbooks**
   - Periodically scan in production to watch for drift

5. **OMS Solution for Alerting & Monitoring**
   - Single security dashboard across DevOps stages

6. **Cloud Risk Governance**
   - Make data-driven improvements to security

---

Make data-driven improvements to security
AzSK – what it covers v. does not cover

What it covers:
- **Control plane** (cloud resource configuration) security
- Drive uniform compliance for subscription and resources for cloud apps across all stages of dev ops
- Works in a decentralized manner
- Enable dev ops teams to do security monitoring of their own apps
- Visibility of security baseline state across the enterprise via extensive telemetry

What it does not cover:
- Data plane configuration checks (AzSK is not ‘agent-based’)
- v1/ASM resources (except Cloud Service). These are deprecated for use at CSE.
- SaaS/O365 security
- Replace SAST/DAST tools (these should carry over from on-prem solutions)
- Network layer things like DLP, IDS, FW, etc.
- Not all Azure services are covered (only top used ones)
Using AzSK at your organization

Level-0: Trying out the AzSK ‘out-of-box’ from aka.ms/devopskit/docs

Level-1: Setup ‘org-specific’ instance of AzSK
  • Simple settings/config changes
  • Pretty much mimic what we have at CSE

Level-2: Customizing AzSK for your environment
  • Leverage integration points for other listeners, custom tags, telemetry, etc.

Level-3:
  • Maintaining your own version (branch) of AzSK (e.g., cost optimization, inventory)
L0: AzSK dataflow (‘preview’/’trial’ use)

What you get:
- Basic rule set
- PS SVTs, SecIntel, CICD, CA, OMS, AI Telemetry
- All with default (central) policy
L1: AzSK – CDN v/s ‘Org-specific’ instance

Visual Studio | Marketplace

PowerShell Gallery

Dev m/c or CICD/CA-VM

Control-JSON

Org Sub

Get-Controls

TM ETL Svc

Telemetry

SQL

Org Sub

App Insights

Power BI

Config, IAM

App Insights

PS Modules

Control + Config

TCP Portal

AzSK Public Sub

Control-CDN

CSV, LOG

Automation

OMS

App Team Sub
Configurability:
- Org-specific settings (e.g., 5 admins, RBAC values, etc.)
- Control settings (e.g., descriptions, severity, filters, etc.)
- Custom report types (e.g., SOX, DR, etc.)
- Other workflows (e.g., attestation expiry period)
Configurability & Extensibility

Configurability:
Org-specific settings (e.g., 5 admins, RBAC values, etc.)
Control settings (e.g., descriptions, severity, filters, etc.)
Custom report types (e.g., SOX, DR, etc.)
Other workflows (e.g., attestation expiry period)

Extensibility – Core Framework
Entirely new SVTs (e.g., new cloud service types)
New control for existing SVTs (e.g., org-specific policy)
CICD tasks for other engines
Secure DevOps Kit for Azure (AzSDK)

Security Report

Subscription Name: MSDN-mprabhu-msft
SubscriptionId: 6bc7464b-1dc0-4141-b32f-57cf4abc4aed
AzSDK Version: 2.4.8
Generated by: AzSDK
Generated on: July 31, 2017 17:45 (UTC)
Requested by: mprabhu11@live.com (User)
Command Executed: Get-AzSDKSubscriptionSecurityStatus -SubscriptionId '6bc7464b-1dc0-4141-b32f-57cf4abc4aed' -GeneratePDF Portrait
Documentation: http://aka.ms/azsdkdocs
FAQ: http://aka.ms/azsdkdocs/faq
Support DL: mailto:ismazsdksup@microsoft.com
Security tests in CICD pipeline - Jenkins
Configurability & Extensibility

Configurability:
- Org-specific settings (e.g., 5 admins, RBAC values, etc.)
- Control settings (e.g., descriptions, severity, filters, etc.)
- Custom report types (e.g., SOX, DR, etc.)
- Other workflows (e.g., attestation expiry period)

Extensibility – Core Framework
- Entirely new SVTs (e.g., new cloud service types)
- New control for existing SVTs (e.g., org-specific policy)
- CICD tasks for other engines
- SecIntel rule templates

Extensibility – Downstream Integration
- Event Hub
- OMS
- Webhook (e.g., Splunk)
- Easy to extend SVT framework
Downstream extensibility - connectors

```
$SubscriptionId = "abb5301a-xxxx-xxxx-9e5f-99baxxxx61f8"
#$SubscriptionId = (Get-AzureRmContext).Subscription.Id

Set-AzSDKEventHubSettings
   -EventHubNameSpace "eventhub-common-01" `
   -EventHubName "common-eventhub" `
   -EventHubSendKeyName "ehSend" `
   -EventHubSendKey "VSqqIritI1f0VXXXXXXXXXxyufk9WnKPSk8="

#Set-AzSDKEventHubSettings -Disable

Set-AzSDKWebhookSettings
   -webHookUrl $webHookUrl
   -authZHeaderValue $authZHeaderValue

Set-AzSDKWebhookSettings
   -WebhookUrl $webHookUrl

#Set-AzSDKWebhookSettings -Disable

Get-AzSDKSubscriptionSecurityStatus -SubscriptionId $SubscriptionId
```
Control results in EventHub

<table>
<thead>
<tr>
<th>PartitionKey</th>
<th>SequenceNumber</th>
<th>Offset</th>
<th>EnqueuedTimeUtc</th>
</tr>
</thead>
<tbody>
<tr>
<td>307</td>
<td>208504</td>
<td>8/13/2017 11:52 AM</td>
<td></td>
</tr>
<tr>
<td>310</td>
<td>207744</td>
<td>8/13/2017 11:52 AM</td>
<td></td>
</tr>
<tr>
<td>306</td>
<td>205864</td>
<td>8/13/2017 11:51 AM</td>
<td></td>
</tr>
<tr>
<td>308</td>
<td>205864</td>
<td>8/13/2017 11:51 AM</td>
<td></td>
</tr>
</tbody>
</table>

Event Text:

```json
{
    "ResourceType": "null",
    "ResourceGroup": "null",
    "Reference": "aka.ms/azsksosttp/sisheal",
    "ResourceId": "null",
    "ResourceIdName": "null",
    "ControlId": "null",
    "ActualVerificationResult": "Passed",
    "Control": [
        "Azure_Subscription_Authz_classic_Resources",
        "SubscriptionName": "MSCT - SECURITY REFERENCE ARCHITECTURE - 01",
        "SubscriptionId": "ab55b19d-020a-4459-9e5f-9bedff29219",
        "FeatureName": "SubscriptionCore",
        "Source": "CISCO",
        "Recommendation": "Migrate each VM/AWS-based resource in your app to a"
    ]
}
```
Controls results in Splunk (via Webhook)

```json
"ResourceId": null,
"ResourceGroup": null,
"Reference": "aka.ms/asdkosstcp/sshealth",
"ResourceId": null,
"ChildResourceName": null,
"ControlStatus": "Passed",
"ActualVerificationResult": "Passed",
"ControlId": "Azure_Subscription_Auth2_Classic_Resources",
"SubscriptionId": "MSFT - SECURITY REFERENCE ARCHITECTURE - 02",
"SubscriptionId": "SubscriptionCore",
"Source": "CC",
"Recommendation": "Migrate each VHAASM-based resource in your app to a corresponding v2/ARM-based resource",
"ControlSeverity": "High",
"TimeTakenInMs": null,
"AttestedBy": null,
"Justification": null
```
Maintaining your own version (branch) of AzSK (e.g., cost optimization, inventory)
Next steps...

Try out the Secure DevOps Kit for Azure!

- [GitHub](https://github.com/azsk)
- Installation guide, docs: [https://aka.ms/devopskit/docs](https://aka.ms/devopskit/docs)
- Controls coverage: [https://aka.ms/devopskit/tcp](https://aka.ms/devopskit/tcp)
- IT Showcase: [https://aka.ms/devopskit/itshowcase](https://aka.ms/devopskit/itshowcase)
- Support: [azsksup@microsoft.com](mailto:azsksup@microsoft.com)
Questions?

Thanks for attending!
Outline of the workshop...

Session0 – Introduction (till here!)

Session1 – Getting Started with AzSK
  Setup & installation
  Subscription and application security scanning
  Understanding scan reports and fixing issues

Session2 – Exploring AzSK (dev-test workflows)
  Walkthrough of various AzSK cmdlets, artifacts
  Using Security IntelliSense to write secure code
  Integrating security into build/release (CICD) pipelines

Session3 – Exploring AzSK (production workflows)
  Customizing for enterprise environments
  Continuous assurance
  Integration with security dashboards for monitoring

Session4 – Close-out
  Next steps
  Validate action items for both teams
Learning objectives for the workshop

• Understand challenges and opportunities for Secure DevOps in Azure
• Hands-on introduction to the powerful capabilities of the Secure DevOps Kit for Azure (AzSK)
• Experience the speed/efficiency/simplicity of AzSK in a "hands on" manner
• Discuss key scenarios where AzSK can be readily used for your org
• Understand advanced use cases of AzSK and (possible) new requirements for AzSK team
• Identify follow-ups, action items and next steps for both teams
Session1 - Getting Started

Setup & installation
Subscription and application security scanning
Understanding scan reports and fixing issues
Secure DevOps Kit for Azure

1. Provision security in subscription
2. Develop securely, spot check security via scripts
3. Deploy securely from VSO build/release pipeline
4. Periodically scan in production to watch for drift
5. Continuous Assurance Runbooks
6. Make data-driven improvements to security

Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)

Security IntelliSense, Security Verification Tests (SVTs)

Telemetry Dashboard
OMS Solution for Alerting & Monitoring
Continuous Assurance Runbooks
CICD Build/Release Extensions

Single security dashboard across DevOps stages
Session2 – Exploring AzSK

Dev-test workflows

- Walkthrough of various AzSK cmdlets, artifacts
- Using Security IntelliSense to write secure code
- Integrating security into build/release (CICD) pipelines
Secure DevOps Kit for Azure

1. Provision security in subscription
2. Develop securely, spot check security via scripts
3. Deploy securely from VSO build/release pipeline
4. Periodically scan in production to watch for drift
5. Single security dashboard across DevOps stages
6. Make data-driven improvements to security
Session 3 – Exploring AzSK

Production workflows

Customizing for enterprise environments
Continuous assurance
Integration with security dashboards for monitoring
Secure DevOps Kit for Azure

1. Provision security in subscription
2. Develop securely, spot check security via scripts
3. Deploy securely from VSO build/release pipeline
4. Periodically scan in production to watch for drift
5. Single security dashboard across DevOps stages
6. Make data-driven improvements to security

- Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)
- Security IntelliSense, Security Verification Tests (SVTs)
- CICD Build/Release Extensions
- Continuous Assurance Runbooks
- OMS Solution for Alerting & Monitoring
- Cloud Risk Governance

Make data-driven improvements to security
AzSK-OSS dataflow (for external users)
Requirements & challenges

Need easy, seamless experience applicable to any enterprise
Cannot have a “consent UI” (must work in CICD, CC)
Possible options need AAD admin consent for app or Graph API permission

someone@contoso.com

orgSeverURL

Contoso-specific JSON
AzSK-OSS assets – runtime

- Visual Studio | Marketplace
- PowerShell Gallery
- Dev Box or CICD/CC-VM
- Control + Config
- App Insights
- Control-JSON
- TCP Portal
- Control-CDN
- Public (SRA06)
- CSV, LOG
- Automation
- OMS
- App Team/BU
- MS/IT Prod
- Get-Controls
- Config, IAM
- Telemetry
- TM ETL Svc
- Power BI
Questions?

Thanks for attending!
Subscription Security  
(Policy, ASC Config, Alerts, RBAC, etc.)

Telemetry Dashboard

Security IntelliSense, Security Verification Tests (SVTs)

OMS Solution for Alerting & Monitoring

CICD Build/Release Extensions

Continuous Assurance Runbooks
Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)

Telemetry Dashboard

OMS Solution for Alerting & Monitoring

Security IntelliSense, Security Verification Tests (SVTs)

CICD Build/Release Extensions

Continuous Assurance Runbooks
Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)

Security IntelliSense, Security Verification Tests (SVTs)

Telemetry Dashboard

OMS Solution for Alerting & Monitoring

CICD Build/Release Extensions

Continuous Assurance Runbooks
Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)

Security IntelliSense, Security Verification Tests (SVTs)

CICD Build/Release Extensions

Continuous Assurance Runbooks

OMS Solution for Alerting & Monitoring

Telemetry Dashboard
Subscription Security
(Policy, ASC Config, Alerts, RBAC, etc.)

- telemetry dashboard
- OMS Solution for Alerting & Monitoring
- continuous assurance runbooks
- CICD Build/Release Extensions
- Security IntelliSense, Security Verification Tests (SVTs)
Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)

Telemetry Dashboard

OMS Solution for Alerting & Monitoring

Continuous Assurance Runbooks

CICD Build/Release Extensions

Security IntelliSense, Security Verification Tests (SVTs)
Subscription Security
(Policy, ASC Config, Alerts, RBAC, etc.)

Security IntelliSense,
Security Verification Tests (SVTs)

Telemetry Dashboard

OMS Solution for Alerting & Monitoring

Continuous Assurance Runbooks

CICD Build/Release Extensions
Subscription Security
(Policy, ASC Config, Alerts, RBAC, etc.)

Security IntelliSense, Security Verification Tests (SVTs)

Telemetry Dashboard

OMS Solution for Alerting & Monitoring

CICD Build/Release Extensions

Continuous Assurance Runbooks
Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)

Security IntelliSense, Security Verification Tests (SVTs)

Telemetry Dashboard

OMS Solution for Alerting & Monitoring

CICD Build/Release Extensions

Continuous Assurance Runbooks
Subscription Security (Policy, ASC Config, Alerts, RBAC, etc.)

Security IntelliSense, Security Verification Tests (SVTs)

Telemetry Dashboard

OMS Solution for Alerting & Monitoring

Continuous Assurance Runbooks

CICD Build/Release Extensions