



**Azure cloud
pentesting stories &
vulnerability research**



Azure cloud pentesting stories

NetSPI is home to many of the top cloud penetration testers in the world. In this notebook, you'll find three stories from NetSPI's security experts, featuring their critical discoveries, unique approaches, and the impact they've made on the security of Azure.

CHAPTER 1

App Registration Certificates Stored in Azure Active Directory – CredManifest CVE

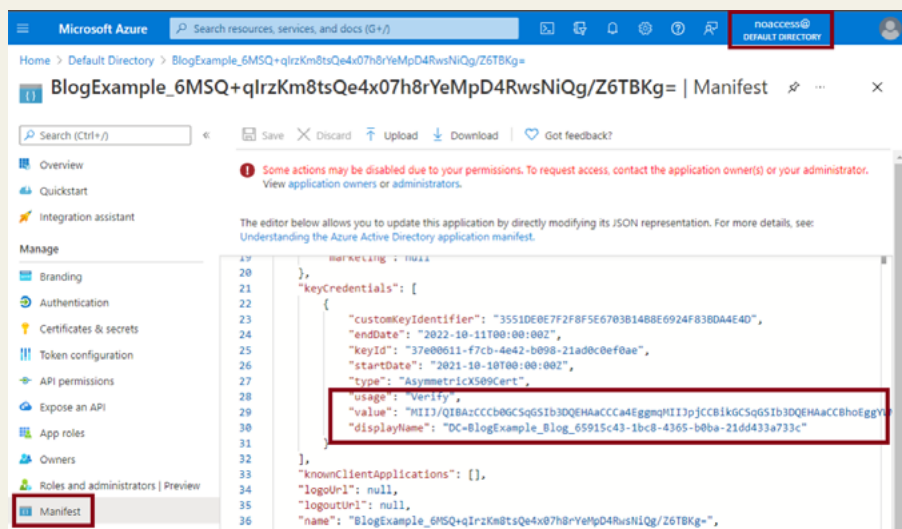
CHAPTER 2

Abusing Azure Hybrid Workers for Privilege Escalation

CHAPTER 3

Azure Privilege Escalation via Cloud Shell

CredManifest: App Registration Certificates Stored in Azure Active Directory



DISCOVERY AND IMPACT

NetSPI VP of Research Karl Fosaaen identified a **misconfiguration in Azure** where Automation Account “Run as” credentials were stored in cleartext in Azure Active Directory (AAD). **This resulted in an impactful privilege escalation, as it would allow any user in this environment to escalate to Contributor of any subscription with an Automation Account.**

[READ THE RESEARCH](#)



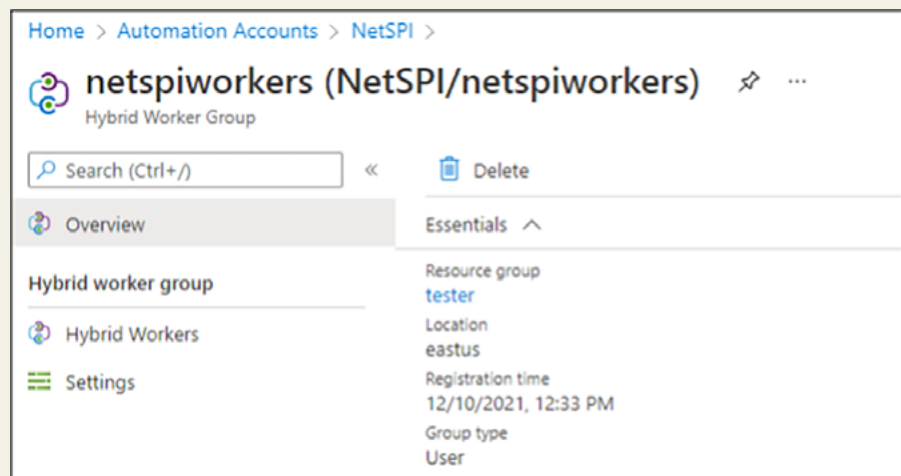
HOW WE FOUND IT

- 1 Identified an issue in the way the Automation Account “Run as” credentials were created when creating a new Automation Account in Azure.
- 2 Manually extract credentials by copying the certificate data out of the manifest and converting it to a PFX file. **We did this with two lines of PowerShell.**
- 3 Import the certificate to our local store using PowerShell in a local administrator session.
- 4 Use the newly installed certificate to authenticate to the Azure subscription as the App Registration.
- 5 With the Directory (Tenant) ID, App (Client) ID, and Certificate Thumbprint values available, run the Add-AzAccount command to authenticate to the tenant.
- 6 Develop PowerShell script to automate extraction.

REMIEDIATION

- We responsibly disclosed the vulnerability to Microsoft who has since deployed updates that prevent cleartext private key data from being stored during application creation and prevents access to private key data previously stored.
- **NetSPI recommends cycling existing Automation Account “Run as” certificates, given the potential exposure of these credentials.**

Abusing Azure Hybrid Workers for Privilege Escalation



DISCOVERY AND IMPACT

When Azure Hybrid Workers are configured to use Automation Account “Run as” accounts, they can **expose the credentials to anyone with local administrator access to the Hybrid Worker**. Since “Run as” accounts are typically subscription contributors, this can lead to **privilege escalation** from multiple Azure Role-Based Access Control (RBAC) roles.

[READ THE RESEARCH](#)



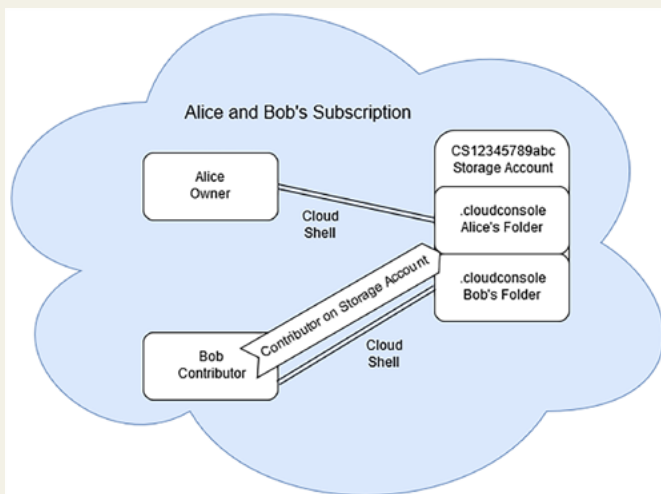
HOW WE DID IT

- 1 Identify Automation Accounts that use Hybrid Workers. **Three options:**
 - a Look at the “Hybrid worker groups” section of an Automation Account in the portal.
 - b Use the Az PowerShell cmdlets to identify the Hybrid Worker groups.
 - c Enumerate the VMs that have the “HybridWorkerExtension” VM extension installed (best option).
- 2 Access Windows VMs in Azure through direct authentication or platform level command execution and extract “Run as” credentials from Hybrid Workers.
- 3 Once admin access is achieved, export the certificate from the worker via RDP access, certmgr, find “Run as” cert, and export to pfx file.
- 4 Copy the file from the Hybrid Worker to use for authentication on another system. We automated the whole process with a **PowerShell script**.
- 5 Authenticate as the “Run as” account using a generated script.

REMEDIATION

- NetSPI responsibly disclosed this vulnerability. Since this issue ultimately relies on an Azure administrator giving a user access to specific VMs (the Hybrid Workers), it’s considered a **user misconfiguration** issue.
- Microsoft updated their documentation to reflect the potential impact of installing the “Run as” certificate on the VMs.
- You could also modify the certificate installation process to mark the certificates as “non-exportable” to help protect them.
- NetSPI recommends against using “Run as” accounts for Automation Accounts and instead **switch to using managed identities on the Hybrid Worker VMs**.

Azure Privilege Escalation via Cloud Shell



DISCOVERY AND IMPACT

Azure Cloud Shell can be a handy way to manage Azure resources, but it can also be a potential source of sensitive data and privilege escalation during a penetration test. By modifying Cloud Shell files, an attacker can execute commands in the sessions of other users. This could lead to **cross-account command execution and privilege escalation**.

READ THE RESEARCH



HOW WE DID IT

- 1 This issue requires **initial access** to a Contributor role in an Azure subscription.
- 2 In a shared subscription, the Contributor has the rights (by default) to download any Cloud Shell IMG file, including files from the owner subscription with administrator access.
- 3 Contributor role downloads IMG file and mounts it in a Linux system.
- 4 Append any attacking commands, **unmount the IMG file**, and upload it back to the Azure Storage Account.
- 5 Select "Overwrite if files already exist" box.
- 6 Once the upload is complete, the Cloud Shell environment is ready for the attack.
- 7 The next Cloud Shell instance launched by an Administrator account in that subscription will run the appended commands.

REMEDIATION

- Both issues (information disclosure and privilege escalation) were submitted to MSRC.
- Because the issues are related to core functionality and designed behavior in Azure (controlling access to storage accounts and file shares, designed behavior for logging) they were not able to make updates.
- Microsoft expanded **guidance on the issue** and suggests users should only grant access to trusted users.
- In the interim, we recommend not using Cloud Shell if you can avoid it.

Meet our cloud pentesting and research leaders



Thomas Elling


Director, Proactive Cloud Security

 <https://www.linkedin.com/in/thomaselling1/>



Karl Fosaaen

VP of Research

 <https://www.linkedin.com/in/karl-fosaaen/>



**Learn more about
our cloud penetration
testing solutions.**



<https://www.netspi.com/security-testing/cloud-penetration-testing>

Keep in touch:



@netspi