Add, Remove and Update Tags for Resources in Azure

Scenario

In the scenario for this hands-on lab, the finance department has reached out to you. They are requesting additional taxonomy information on a recent Azure bill, including who created the resources, which department budget should be used for the resources, and if the resources are necessary for running business critical systems. If there are any non-essential business systems, they ask that you signify that in some way.

About this project

This project focused on using Azure Cloud Shell and CLI commands to manage resources through tagging. I practiced listing resource groups, virtual machines, and virtual networks to review their properties.

I then applied, updated, removed, and overwrote tags to organize resources with metadata such as Environment, Department, and CreatedBy.

Through this process, I learned how tagging improves visibility, simplifies management, and supports governance in cloud environments.

I also found that tags can be used not only for organization but also for operational tasks like marking resources for deletion.

Skill Set

🌐 Azure Administration

💻 Azure Cloud Shell

⌨️ Azure CLI Proficiency

🏷️ Tagging & Governance

📂 Resource Management

🚦 Operational Awareness

🧩 Problem Solving

Contact Details

ikuesan1503@gmail.com

07561 735514

https://github.com/Ikedrew?tab=repositories

www.linkedin.com/in/andrew-ikuesan-525654183

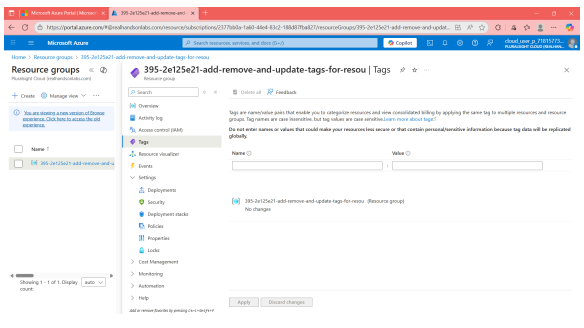
Azure Resource Management with Cloud Shell and CLI

1st Stage

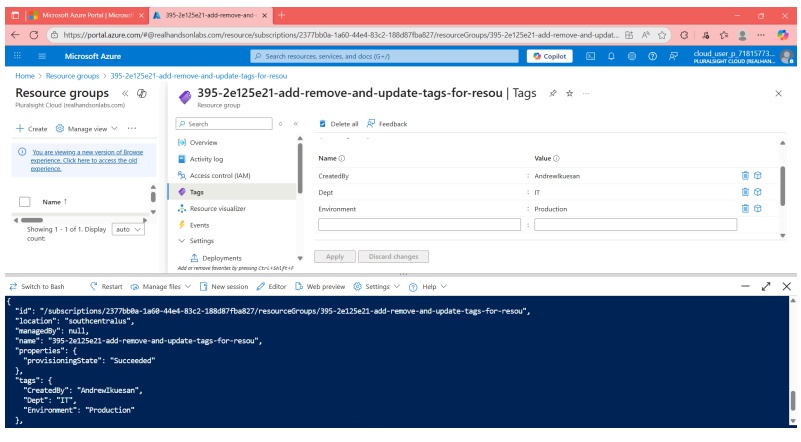
I logged into the Azure Portal using the credentials provided in the lab instructions. From there, I opened the Cloud Shell through the portal and selected PowerShell as the environment. I continued without a storage account, applied the available subscription, and launched the shell. Finally, I switched the session to Bash in order to run Linux-based commands for the lab.

2nd Stage

I listed the available resource groups in Azure using the `az group list` command and copied the group name. I then updated the resource group by applying tags through the `az group update` command, assigning values for Environment, Department, and CreatedBy. This ensured the resource group was organized with metadata for easier management and identification.



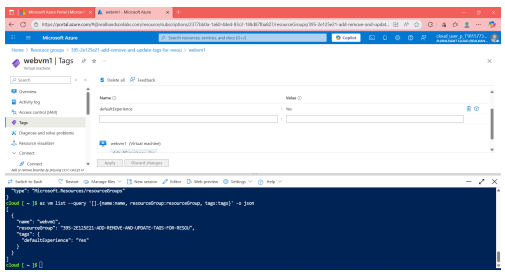
This shows the tags section of the resource group before I used PowerShell to add new tags



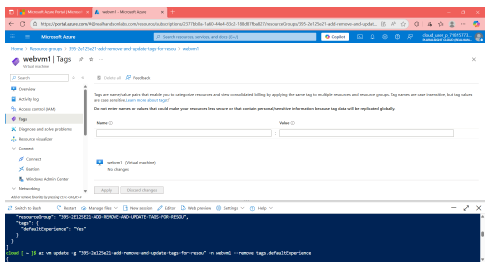
These are the resource group tags I added: CreatedBy, Department, and Environment

3rd Stage

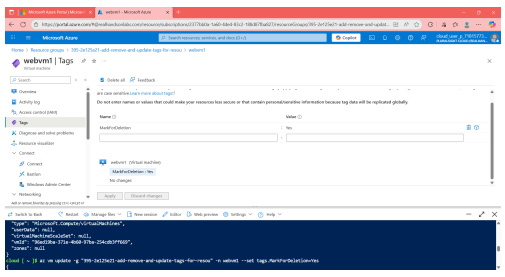
In the Cloud Shell, I listed the existing virtual machines using the `az vm list` command to view their names, resource groups, and tags. I then removed the existing tags from webvm1 by running the `az vm update` command with the `--remove` option. Finally, I marked webvm1 for deletion by applying a new tag using the `--set` option, which flagged it for removal while keeping the process organized.



Queried VM list in PowerShell and retrieved `webvm1` with its resource group and tag



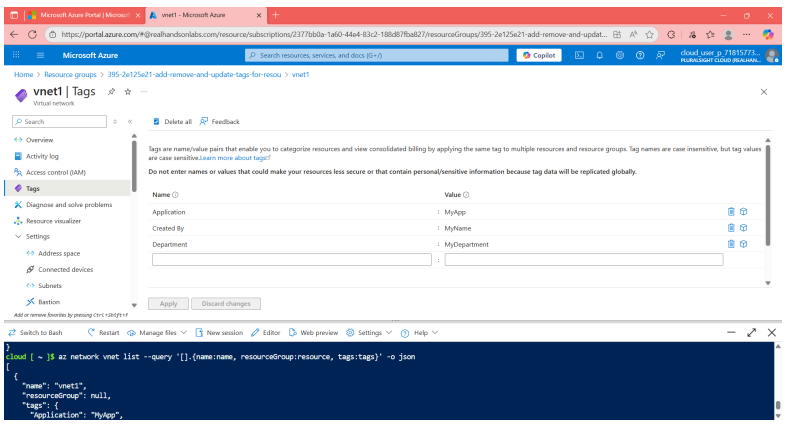
Next, I updated `webvm1` to remove the existing tag



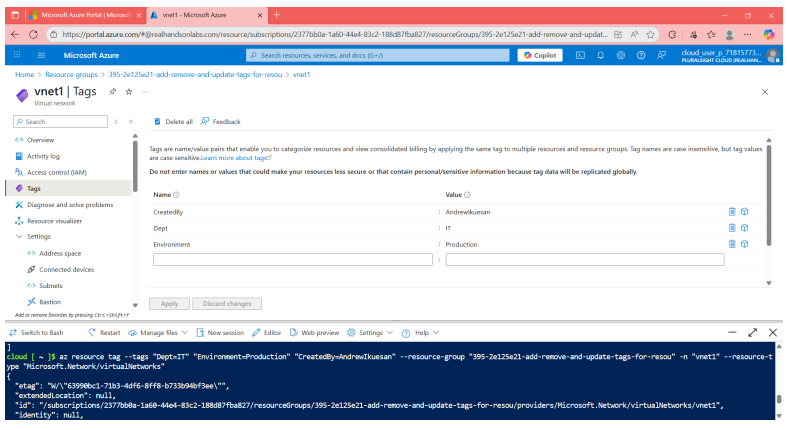
Finally, I set a new tag on `webvm1` to mark it for deletion.

4th Stage

In the Cloud Shell, I listed the existing virtual networks using the `az network vnet list` command to view their names, resource groups, and current tags. I then overwrote the existing tags for the virtual network by running the `az resource tag` command, assigning values for Department, Environment, and CreatedBy to ensure proper organization and identification within the resource group.



I queried the VNet list to view its name, resource group, and tags, which returned the app name, creator, and department



I replaced the existing tags with CreatedBy, Department, and Environment

💡**Key Takeaways:**

- Learned how to query Azure resources (VMs, VNets, and resource groups) using PowerShell and CLI.
- Gained hands-on experience adding, removing, and updating tags to support governance and cost tracking.
- Understood how tags improve visibility, accountability, and resource management across departments.