

# Azure Vulnerability Lab Scanner

This project showcases practical cloud security operations by implementing a complete vulnerability management process in Microsoft Azure, including asset deployment, threat discovery, credentialled scanning, vulnerability remediation, and post-fix verification using OpenVAS.

## Prerequisites

- Computer with Internet access
- Azure Account (Free Subscription may be possible)
  - Create your Free Azure Account [here](#)
  - Log in to your Azure Account [here](#)

## The Vulnerability Scanning Process

**Step 1**



Identification  
and Inventory

**Step 2**



Detection  
and Analysis

**Step 3**



Remediation  
and Continuous  
Monitoring

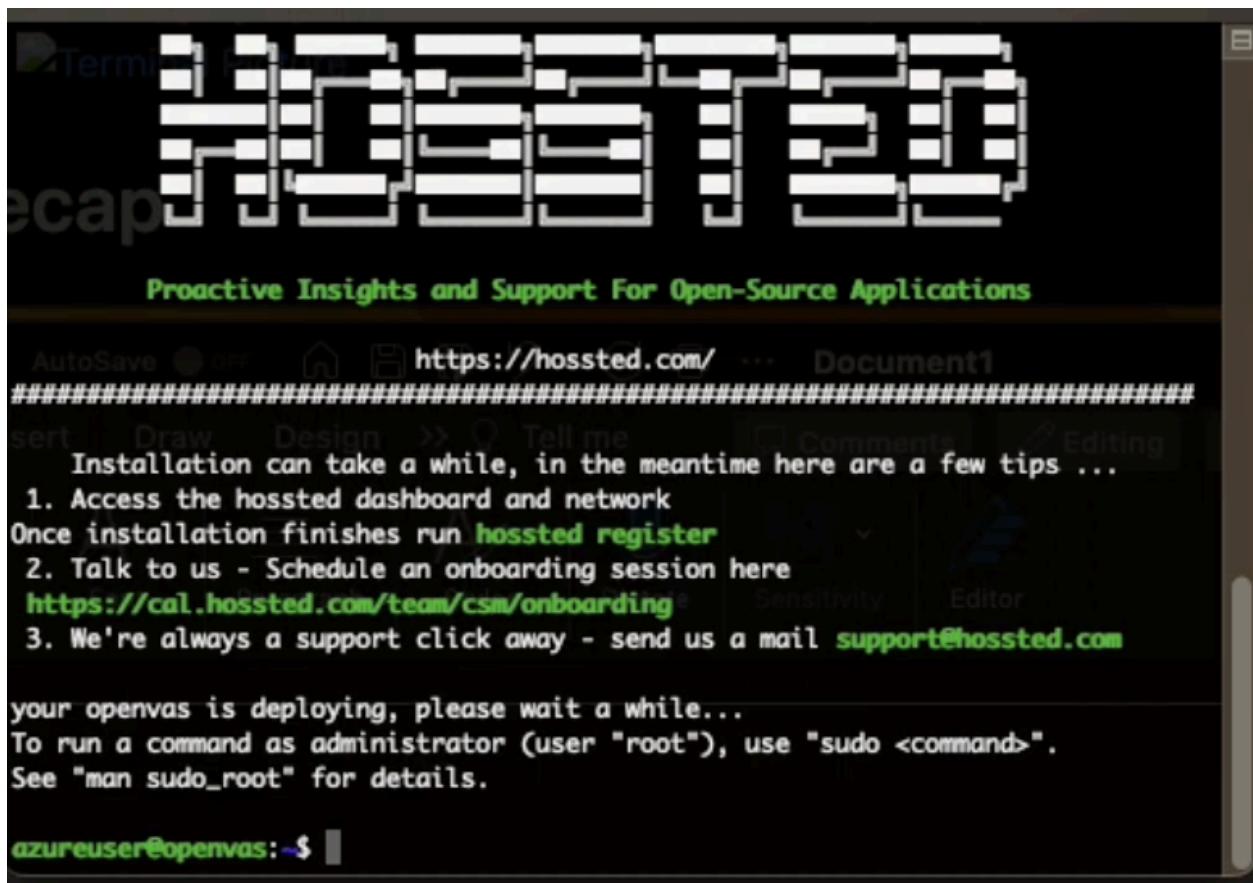
# Step 1: Prepare Vulnerability Management Scanner

1. Go to [Azure Portal](#).
2. Navigate to the Marketplace and search for "OpenVAS secured and supported by HOSSTED."
3. Choose "Start with a pre-set configuration" and select the weakest configuration.

The screenshot shows the Azure Marketplace configuration interface for a pre-set configuration. At the top, there's a header with a cloud icon and the text "Choose recommended defaults that match your workload". Below it, a note says "configurations at any time." A "Select a workload environment" section contains two boxes: "Dev/Test" (with "Boot diagnostics" checked) and "Production" (with "Boot diagnostics", "High availability", and "Azure backup (where available)" checked). Under "Select a workload type", there are three boxes: "General purpose (D-Series)" (with "Example sizes" (DS2\_v2, DS3\_v2), "Fast CPUs with optimal CPU-to-memory configuration", and "Workload types" (Enterprise applications, relational databases, analytics)), "Memory optimized (E-Series)" (with "Example sizes" (E2s\_v3, E4s\_v3), "High memory-to-core ratio optimized for heavy in-memory applications", and "Workload types" (SAP HANA, SQL Hekaton, other large in-memory workloads)), and "Compute optimized (F-Series)" (with "Example sizes" (F2s\_v2, F4s\_v2), "High CPU-to-memory ratio optimized for compute intensive workloads", and "Workload types" (Batch processing, web servers, gaming)).

4. Continue to create the VM:
  - Resource Group: Vulnerability-Management
  - VM Name: OpenVAS (Note the region and Vnet – consider East US 2)
  - Security: Standard
  - Authentication: Username → azureuser / Password → Cyberlab1234!
  - Monitoring: Disable Boot Diagnostic (not needed)
  - Management: Disable auto-deploy
5. Create the VM.
6. Once created, SSH into it using PowerShell (Windows) or Terminal (MacOS) with the credentials you created.

```
sudo ssh username@ip
```



7. Wait for the deployment to complete, and it should display the web app URL and default username and password.

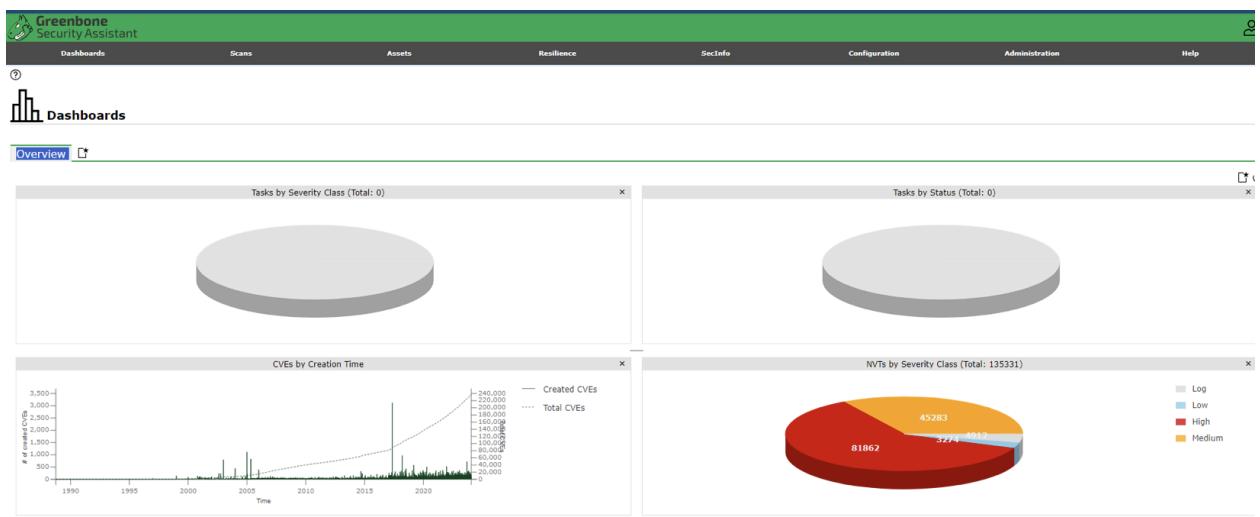
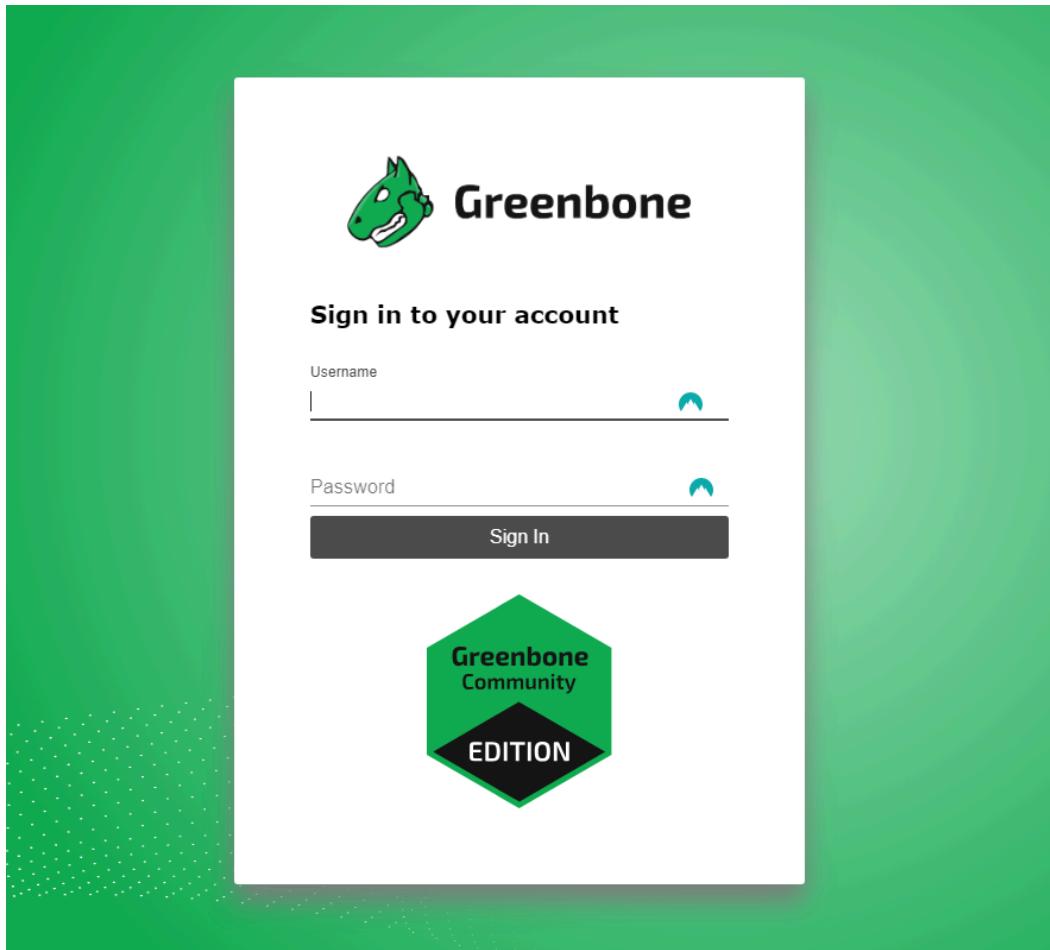
◦ Management: Disable auto-deploy

Broadcast message from root@openvas (somewhere) (Sun Sep 3 07:04:01 2023):

Create the VM.

Username is admin and password is zachu4eeX9

8. Copy the URL and log in to the web app. If it doesn't work, try using admin/admin.



## Step 2: Create Client VM and Make it Vulnerable

The screenshot shows the 'Create a virtual machine' wizard in the Microsoft Azure portal. The 'Basics' tab is selected. The 'Project details' section shows a subscription named 'Azure subscription 1' and a resource group named 'Vulnerability-Management'. The 'Instance details' section includes a virtual machine name 'Win10-Vulnerable', a region '(US) East US 2', availability options ('Availability zone' and 'Zones 1'), and a security type 'Standard'. The 'Image' section shows 'Windows 10 Pro, version 22H2 - x64 Gen2' selected. A note indicates that this image is compatible with additional security features.

1. Go to [Azure Portal](#).
2. Search for Virtual Machines and create a new Virtual Machine.
3. Configure the VM:
  - Resource Group: Vulnerability-Management
  - VM Name: Win10-Vulnerable
  - Region: Same as the OpenVAS VM (East US 2)
  - Virtual Network: Same as OpenVAS
  - Image: Windows 10 Pro
  - Size: Any size with 2 vCPUs
  - Username: azureuser / Password: Cyberlab1234!
  - Networking: Same Vnet as OpenVAS
4. Create the VM.
5. Once created, ensure you can RDP into it with the credentials.
  - MacOS needs an RDP app to access



## 6. the VM vulnerable by:

- Disabling the Windows Firewall (Allows Vulnerability Scanner to Scan the VM without dropping traffic)
  - Start > wf.msc > Firewall Properties
    - Domain Profile > Off
    - Private Profile > Off
    - Public Profile > Off
- Installing old software on Win10-Vulnerable VM! :

<https://drive.google.com/drive/u/2/folders/1n83ilCjZWZulbDdYnUe9wQPK2buY47U>

- Old Version of Firefox: Firefox Setup 97.0b5
- Old Version of VLC Player: vlc-1.1.7-win32

- Old Version of Adobe Reader:  
10.0\_AdbeRdr1000\_en\_US\_1\_

7. Restart the VM.

## Step 3: Configure and Run Unauthenticated Scan with OpenVAS

1. Log in to OpenVAS.
2. Navigate to Assets → Hosts → New Host.
3. Add the Client VM PRIVATE IP Address. -Found in Azure: VM > Win10-Vulnerable Properties > Private IP

Virtual machine		Networking	
Computer name	Win10-Vulnerable	Public IP address	20.172.165.150 ( Network interface win10-vulnerable27 )
Operating system	Windows	Public IP address (IPv6)	-
Image publisher	MicrosoftWindowsDesktop	Private IP address	10.0.0.5
Image offer	Windows-10	Private IP address (IPv6)	-

1. Create a new Target from the Host, naming it "Azure Vulnerable VMs."
2. Take note of the credentials; SMB credentials will be added later.
3. Create a new Task:
  - Name & Comment: "Scan - Azure Vulnerable VMs"
  - Scan Targets: "Azure Vulnerable VMs"
4. Save the Task.
5. Start the "Scan - Azure Vulnerable VMs" Task.
6. Take note of the scan status.
7. After the scan finishes, click the date under "Last Report" to view the results.
8. Examine the "Results" tab. Note that some vulnerabilities may not appear due to the unauthenticated scan.

## Results

- The scan did not find the vulnerable installed apps since it was not an authenticated scan

 Report: Sun, Sep 3, 2023 3:00 AM UTC Done

ID: ea4b79a

Information	Results (3 of 32)	Hosts (1 of 1)	Ports (2 of 4)	Applications (0 of 0)	Operating Systems (1 of 1)	CVEs (2 of 2)	Closed CVEs (7 of 7)	TLS Certificates (1 of 1)	Error Messages (1 of 1)	User Tags (0)
-------------	----------------------	-------------------	-------------------	--------------------------	-------------------------------	------------------	-------------------------	------------------------------	----------------------------	------------------

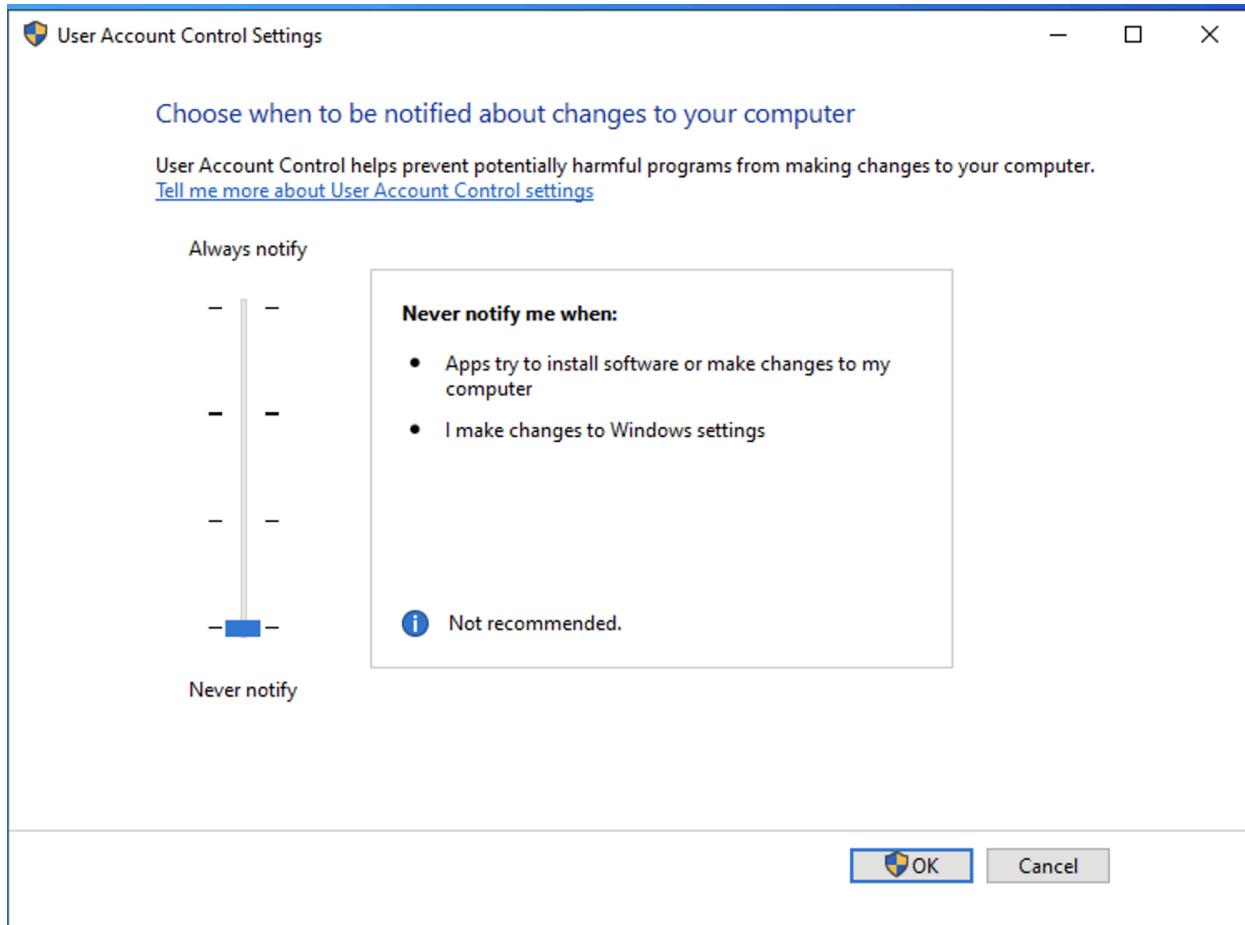
Vulnerability	Severity ▾	QoD	Host	
			IP	Name
DCE/RPC and MSRPC Services Enumeration Reporting	5.0 (Medium)	80 %	10.0.0.5	win10-v
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	4.3 (Medium)	98 %	10.0.0.5	win10-v
ICMP Timestamp Reply Information Disclosure	2.1 (Low)	80 %	10.0.0.5	win10-v

(Applied filter: apply\_overrides=0 levels=hml rows=100 min\_qod=70 first=1 sort-reverse=severity)

## Step 4: Configure Authenticated Scans (Within VM)

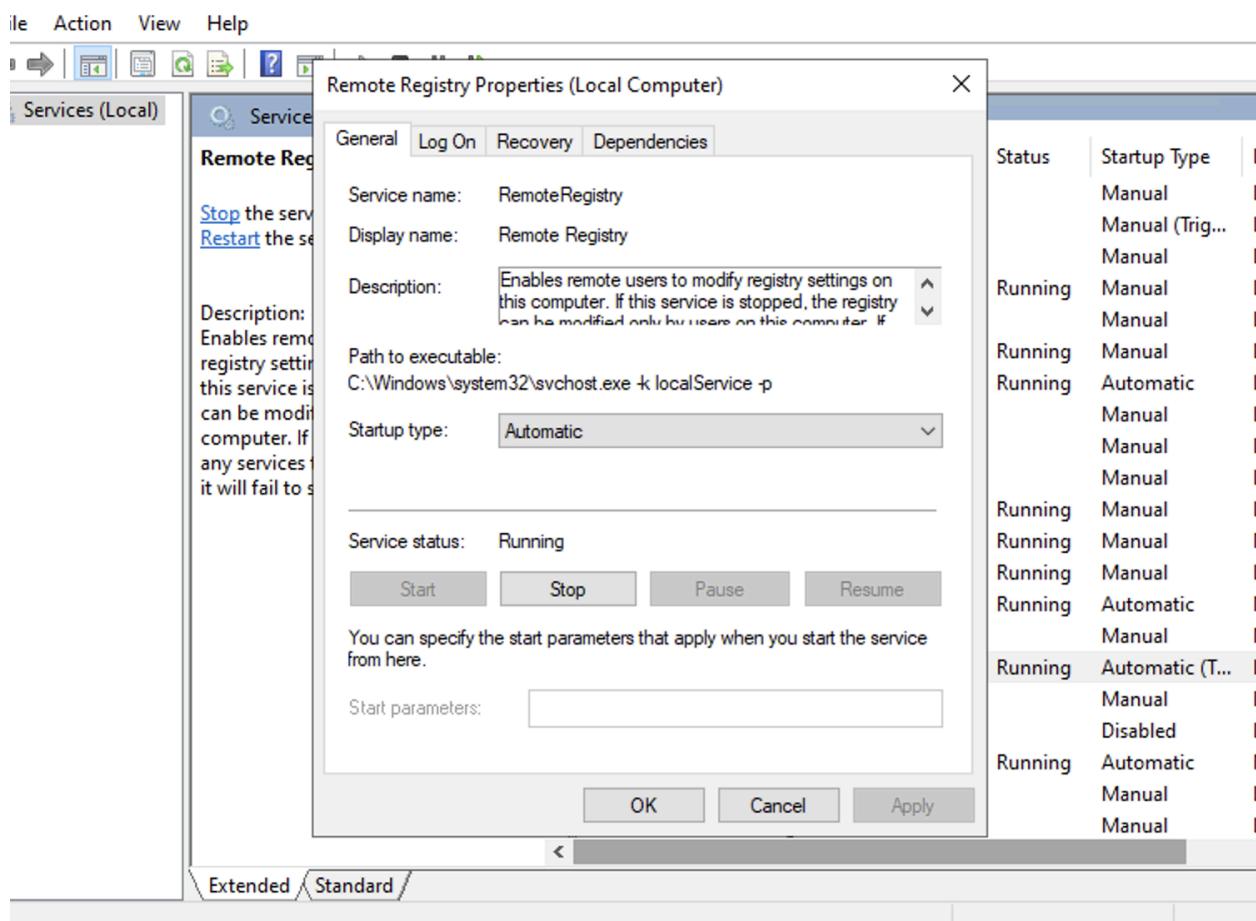
On the vulnerable Windows VM, perform the following configurations:

- Disable Windows Firewall
- Disable User Account Control
  - Start > "user account control" > set to "never notify" -



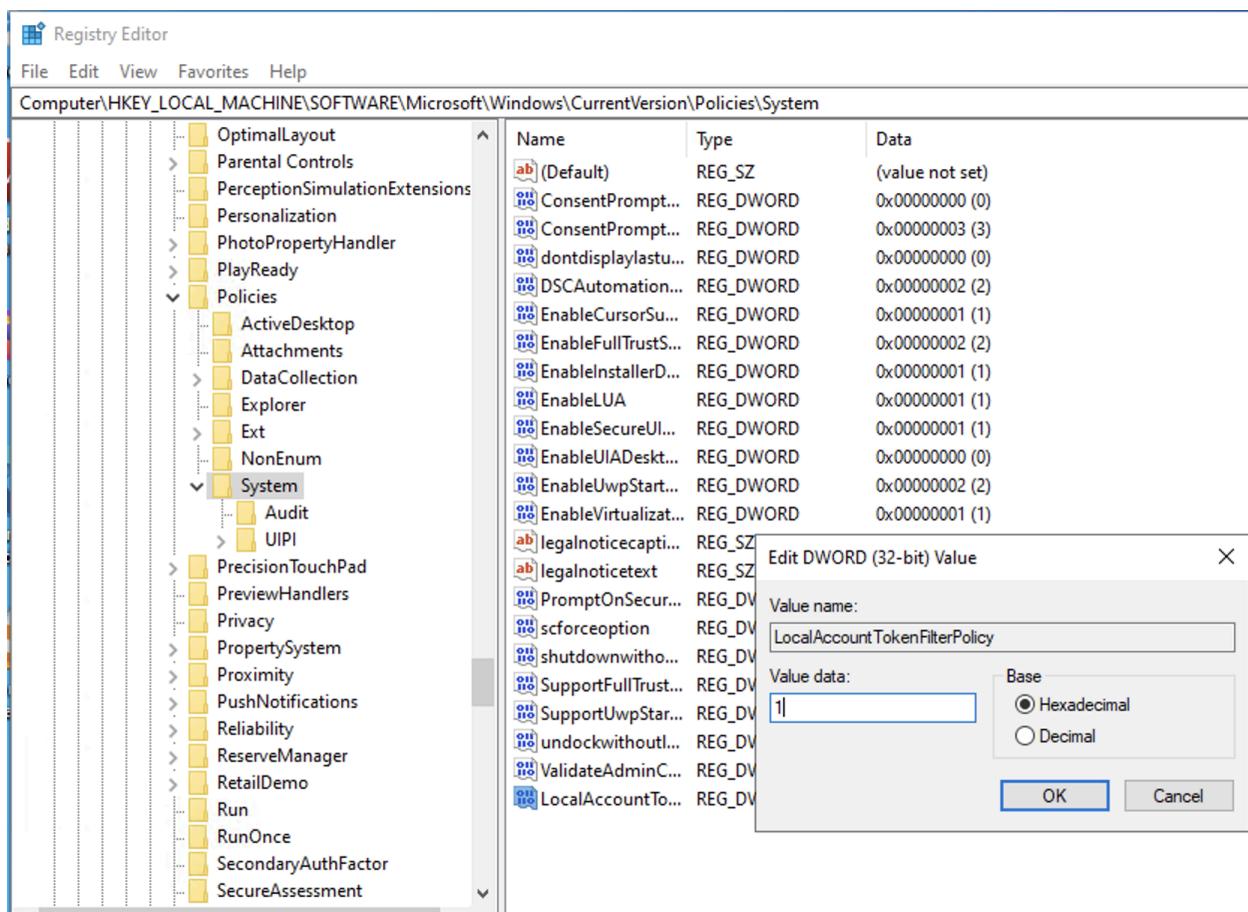
## Enable Remote Registry

- Start > services.msc
  - Search for: Remote Registry > Properties > Startup type: Automatic > Apply > Start -



## Set Registry Key (LocalAccountTokenFilterPolicy: 1)

- Start > regedit > HKEY\_LOCAL\_MACHINE > Open SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
- Create a new DWORD (32-bit) value with the following properties:
  - Name: LocalAccountTokenFilterPolicy
  - Value: 1 -



- Restart VM

## Configure Authenticated Scans (OpenVAS)

1. In OpenVAS, go to Configuration → Credentials → New Credential.
2. Name / Comment: "Azure VM Credentials."
3. Allow Insecure Use: Yes
4. Username: azureuser
5. Password: Cyberlab1234!

New Credential

Name	Azure VM Credentials
Comment	Azure VM Credentials
Type	Username + Password ▾
Allow insecure use	<input checked="" type="radio"/> Yes <input type="radio"/> No
Auto-generate	<input type="radio"/> Yes <input checked="" type="radio"/> No
Username	azureuser
Password	*****

Cancel Save

6. Save.
7. Go to Configuration → Targets → Clone the Target previously created.
8. Name / Comment: "Azure Vulnerable VMs - Credentialed Scan."
9. Ensure the Private IP is still accurate.
10. Credentials → SMB → Select the "Azure VM Credentials" created earlier.

Edit Target Azure Vulnerable VMs Clone 1

Name	Azure Vulnerable VMs Credentialed Scan
Comment	Azure Vulnerable VMs Credentialed Scan
Hosts	<input checked="" type="radio"/> Manual 10.0.0.5 <input type="radio"/> From file Choose File No file chosen
Exclude Hosts	<input checked="" type="radio"/> Manual <input type="radio"/> From file Choose File No file chosen
Allow simultaneous scanning via multiple IPs	<input checked="" type="radio"/> Yes <input type="radio"/> No
Port List	All IANA assigned TCP ▾
Alive Test	Scan Config Default ▾
Credentials for authenticated checks	
SSH	-- ▾ on port 22
SMB	Azure VM Credentials ▾
<input type="button" value="Cancel"/>	<input type="button" value="Save"/>

11. Save.

## Step 5: Execute Authenticated Scan against Vulnerable Windows VM

1. In OpenVAS, go to Scans → Tasks.
2. Clone the "Scan - Azure Vulnerable VMs" Task and edit it:
  - Name / Comment: "Scan - Azure Vulnerable VMs - Credentialed."
  - Targets: Azure Vulnerable VMs - Credentialed Scan

Edit Task Scan - Azure Vulnerable VMs Clone 1

Name	Scan - Azure Vulnerable VMs - Credentialaled
Comment	Scan - Azure Vulnerable VMs - Credentialaled
Scan Targets	Azure Vulnerable VMs Credentialaled Sca ▾ <span style="color: red;">*</span>
Alerts	<input type="button" value="Azure Vulnerable VMs Credentialaled Scan"/>
Schedule	-- ▾ <input type="checkbox"/> Once <span style="color: red;">*</span>
Add results to Assets	<input checked="" type="radio"/> Yes <input type="radio"/> No
Apply Overrides	<input checked="" type="radio"/> Yes <input type="radio"/> No
Min QoD	70 <span style="border: 1px solid #ccc; padding: 2px;">%</span>
Alterable Task	<input type="radio"/> Yes <input checked="" type="radio"/> No
Auto Delete Reports	<input checked="" type="radio"/> Do not automatically delete reports <input type="radio"/> Automatically delete oldest reports but always keep newest <input type="text" value="5"/> reports
Scanner	OpenVAS Default ▾
Scan Config	Full and fast ▾

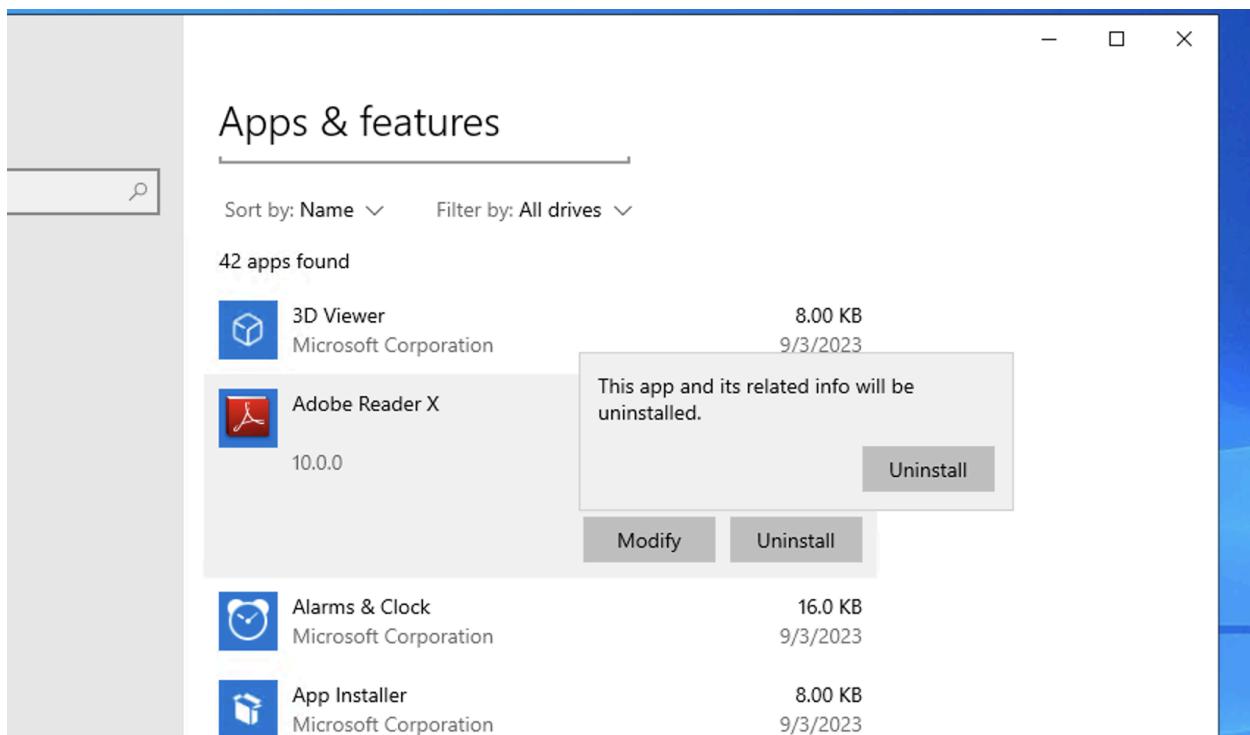
Cancel Save

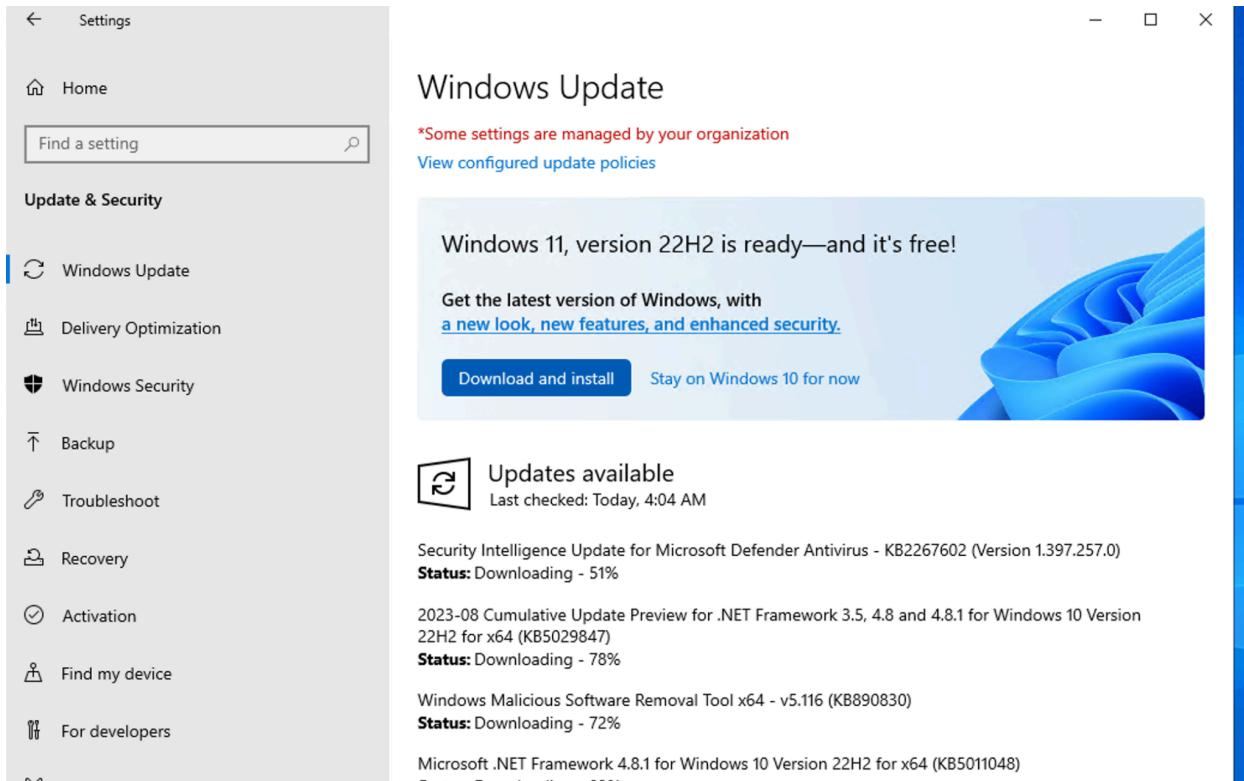
3. Save.
4. Launch the new Credentialaled Scan and wait for it to finish (this may take longer).
5. After the credentialaled scan finishes, check SMB Login under "Results."
6. Inspect individual vulnerabilities, including critical findings related to outdated software.
7. Remove the filter (upper right).

Vulnerability	Severity ▾	QoD	Host	
			IP	Name
Mozilla Firefox Security Update(mfsa_2022-51_2022-53)-Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Adobe Reader Sandbox Bypass Vulnerability (Aug 2014) - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Mozilla Firefox Security Updates(mfsa2022-24) - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Mozilla Firefox Security Update(mfsa_2022-09)- Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Mozilla Firefox Security Update(mfsa2022-33) - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Mozilla Firefox Security Update(mfsa2022-28) - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Adobe Reader Multiple Vulnerabilities - 01 May15 (Windows)	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Adobe Reader Multiple Vulnerabilities - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Adobe Reader Multiple Vulnerabilities April-2012 (Windows)	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Adobe Reader Multiple Vulnerabilities-01 Dec14 (Windows)	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Adobe Reader Multiple Vulnerabilities-01 Sep14 (Windows)	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Adobe Reader/Acrobat 'UDF' Component Memory Corruption Vulnerability (APSA11-04, APSB11-30) - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Mozilla Firefox Security Updates(mfsa2022-20) - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Mozilla Firefox Security Updates(mfsa2022-16) - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net
Mozilla Firefox Security Updates(mfsa2022-19) - Windows	10.0 (High)	97 %	10.0.0.5	win10-vulnerable.internal.cloudapp.net

## Step 6: Remediate Vulnerabilities

1. Log back into your Win10-Vulnerable VM.
2. Uninstall Adobe Reader, VLC Player, and Firefox.
3. Update Windows





#### 4. Restart VM

## Step 7: Verify Remediations

1. Re-initiate the "Scan - Azure Vulnerable VMs - Credentialled" scan and observe the results.

Report: Sun, Sep 3, 2023 4:12 AM UTC      Done

Filter:

ID: 7a69ebb9-0c78-4374-a3de-cee294c204f9      Created: Sun, Sep

Information	Results (7 of 54)	Hosts (1 of 1)	Ports (2 of 5)	Applications (15 of 15)	Operating Systems (1 of 1)	CVEs (4 of 4)	Closed CVEs (2644 of 2644)	TLS Certificates (1 of 1)	Error Messages (1 of 1)	User Tags (0)
<b>Vulnerability</b>										
Windows IExpress Untrusted Search Path Vulnerability										
Windows IExpress Untrusted Search Path Vulnerability										
Microsoft Windows HID Functionality (Over USB) Code Execution Vulnerability (Jan 2011)										
Microsoft Windows HID Functionality (Over USB) Code Execution Vulnerability (Jan 2011)										
DCE/RPC and MSRPC Services Enumeration Reporting										
SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection										
ICMP Timestamp Reply Information Disclosure										

(Applied filter: apply\_overrides=0 levels=hml rows=100 min\_qod=70 first=1 sort-reverse=severity)

## Summary:

I built a cloud-based security lab in **Microsoft Azure** to demonstrate the complete **vulnerability management lifecycle**. The project involved deploying **OpenVAS**, performing both **unauthenticated and authenticated vulnerability scans** against a purposely vulnerable Windows VM, analyzing critical findings, executing **remediation actions**, and validating fixes through re-scanning. This lab showcases hands-on experience with **cloud security operations, risk assessment, remediation, and vulnerability verification**.