Azure Compute Instance Setup

Demo

Instructor: In this demonstration we are going to setup an Azure Compute instance, protect it with time-based access controls, and install several agents to improve the security posture of that resource.

First, we will create a new virtual machine, along with a new resource group. InstanceGroup1. ComputeInstance1.

Let's select a Windows 10 operating system and the option to use an Azure spot Instance. This saves us a couple bucks right now, but affords Azure the options to reclaim our resources based on other demands, so be careful when using this option.

We'll select the smallest instance available for our test VM, and set our username and password.

We are going to disable all inbound network access, because we are instead going to use a time-based ACL later.

We will select the default disk options, and again ensure we are not enabling any inbound ports on the networking tab.

We are going to select the default monitoring options. But this one here is interesting. There is an option for automatic patching for Windows VMs, although it is not completely ready at the time of this recording.

On the Advanced tab we can choose to install additional extensions. These pre-approved add-ons support additional monitoring, automation, malware protection, and vulnerability assessment options.

We'll return, fly through the tags selection page, review and validate our configuration, before creating.

This will take a couple minutes. Once complete, let's take a look at the just-in-time access feature. This requires an upgrade to Security Center for us to use. We'll do this upgrade, but we won't install any additional agents at this time.

Once ready, we can see the Just-in-time access option available to us on

the Configuration tab. When we select it, it takes us back to Security Center to configure.

We will select our instance, and then Request Access. Toggle the off switch to on for RDP, TCP/3389. Allow only my IP, the console is reading the IP address I'm using to connect into Azure right now. And then we can select the duration of this ACL. Finally, we click Open Ports, and if we go to the networking section for our instance, we see the new rule that has been added.

Similarly, if we click connect, we can select which protocol we would like to use to connect to our instance, along with its public IP. We will select My IP again, and Request Access. We can download this RDP config right from here as well.

And if we jump back to the Networking tab, you will see that we have yet another access rule created. These are the exact same, which demonstrates that we can request our time-based access in multiple ways.

For giggles, let's go ahead and connect to verify our access. And once logged in, we will just verify external connectivity by pinging google.

We will have a number of default logging options enabled in Azure, but if we want to easily access Security Events also, we need to take a few extra steps. First, we need to create

a new Log Analytics workspace. We'll use the pay-as-you-go option, no tags, review, and finally create.

Once that is completed, we need to ensure the machines within our resource group are connected to our Log Analytics workspace. That will take just a few minutes.

We also want to install the Azure Security Center Agents. This is required for us to review Security Events that are part of the Windows Event Viewer. This requires us to upgrade Security Center from the Standard tier so we can install this and other agents.

Once the dust settles from those jobs, we can get into View Designer within our Log Analytics Workspace. There are a plethora of options here so we are going to skip ahead to a simple example.

On the left here you will notice a list of tables. This is the database of information available to you. LogManagement and SecurityCenterFree have some useful information, but the Security and Audit one does not appear unless we have the upgraded Security Center and the necessary agents installed.

We will navigate through that one to find an example query. Virtual Machines. Windows Failed Logins. Run.

We just started our instance, and do not have any failed logins yet, but

this could be used investigate an incident such as a brute force login attempt. And further queries could be used to identify rogue processes running on the virtual machine.

Now we can pop back over to Security Center. There are a number of findings that we can address right away. We see we have 4 total recommendations. The first one on our list is related to vulnerability assessment, and there is a quick fix option.

Here we can select our instance, click Remediate, and select the vulnerability assessment agent we would like installed. Azure incorporates a Qualys agent into their standard tier, but you can elect to use a different agent if you have one available. Proceed.

Remediate. And while that is working, we will go back to our dashboard, and find the next one to evaluate. We will not address the disk encryption issue at this time. However, we will follow through on this endpoint protection finding. Select our instance. Select Microsoft Antimalware. Select Create to install the extension.

We will accept most of the defaults but schedule a weekly scan on Saturdays. OK.

And then jump back to our dashboard. Drilling into some of these other findings, we can see that things like DDoS protection are

evaluated, and those checks pass based on Azure's inherent controls.

Underneath security best practices there is another agent we can enable for network traffic data collection, so let's do that now.

In the notes section you will see some dependencies like requiring the Microsoft Monitoring Agent. We will select our computer, and then Remediate. And confirm again.

Once all of these agents are properly installed, we can verify their status by going to our Compute instance. And when we click on extensions, you will see the various agents, which we have enabled to help enhance our overall security posture.