Chaos Engineering Scenario: Leverage Chaos Studio to test an availability zone failure

Introduction

You can use a chaos experiment to verify that your application is resilient to failures by causing those failures in a controlled environment. In this guide, you will cause an availability zone failure on a Linux virtual machine using a chaos experiment and Azure Chaos Studio. Running this experiment can help you defend against service interruption in the event of a zonal outage.

Prerequisites

Before proceeding, you should understand:

- Regions and availability zones (https://dev.azure.com/msazure/AzureWiki/_wiki/wikis/AzureWiki.wiki/59055/Availability-Zones)
- Service level indicators (SLIs) and service level objectives (SLOs) (https://eng.ms/docs/quality/slos-slis)
- The Azure Quality Program (https://eng.ms/docs/quality/program-overview)

Tools

- An Azure subscription (https://docs.microsoft.com/azure/guides/developer/azure-developer-guide#understanding-accounts-subscriptions-and-billing). Create a subscription in AIRS (https://azuremsregistration.microsoft.com/Default.aspx) before you begin.
- An application that runs on virtual machines in Azure Canary regions and follows mandatory safe deployment practices (https://eng.ms/docs/quality/zero-self-inflicted-sev1s/safedeploy).

Scenario background

Availability zones allow you to digitally access physical locations with data centers and specify geographic regions for compliance boundaries. Deploying an application to multiple availability zones allows it to remain resilient to data center failure, or zonal failure, because each zone has its own isolated network and power. When one availability zone fails, the load balancer should redistribute the deployed application to other availability zones without affecting performance.

Service level indicators (SLIs) are metrics such as availability, latency, throughput, and error rate that are used to analyze service quality and reliability. SLIs are the target values for your service level objectives (SLOs), or what the customer expects from a service's performance. You can use SLIs and SLOs when evaluating the effectiveness of your availability zone failure response.

Scenario goal

In this scenario, you will:

- Understand the relationship between availability zones, data centers, and load balancers, and how they are accessed using Chaos Studio.
- Identify and use key metrics to formulate an experiment hypothesis.
- Create an experiment that tests the performance of a deployed application when an availability zone, simulated by virtual machines, shuts down.
- Interpret experiment results to assess and potentially reformulate your created hypothesis.

Establish a hypothesis

Establishing a hypothesis is critical before beginning an experiment. Without a hypothesis, it is difficult to understand what to test or how to interpret any results.

For this scenario, create a hypothesis that addresses both zone down and observability expectations. If an availability zone fails, what do you expect to happen, and how do you expect to receive the results?

To establish a hypothesis, ask questions relevant to the scenario. For example, by running this experiment, what do you expect to happen given your specific application setup, SLI, and SLO? What does a healthy result look like? What is your failure tolerance? What metrics are you assessing?

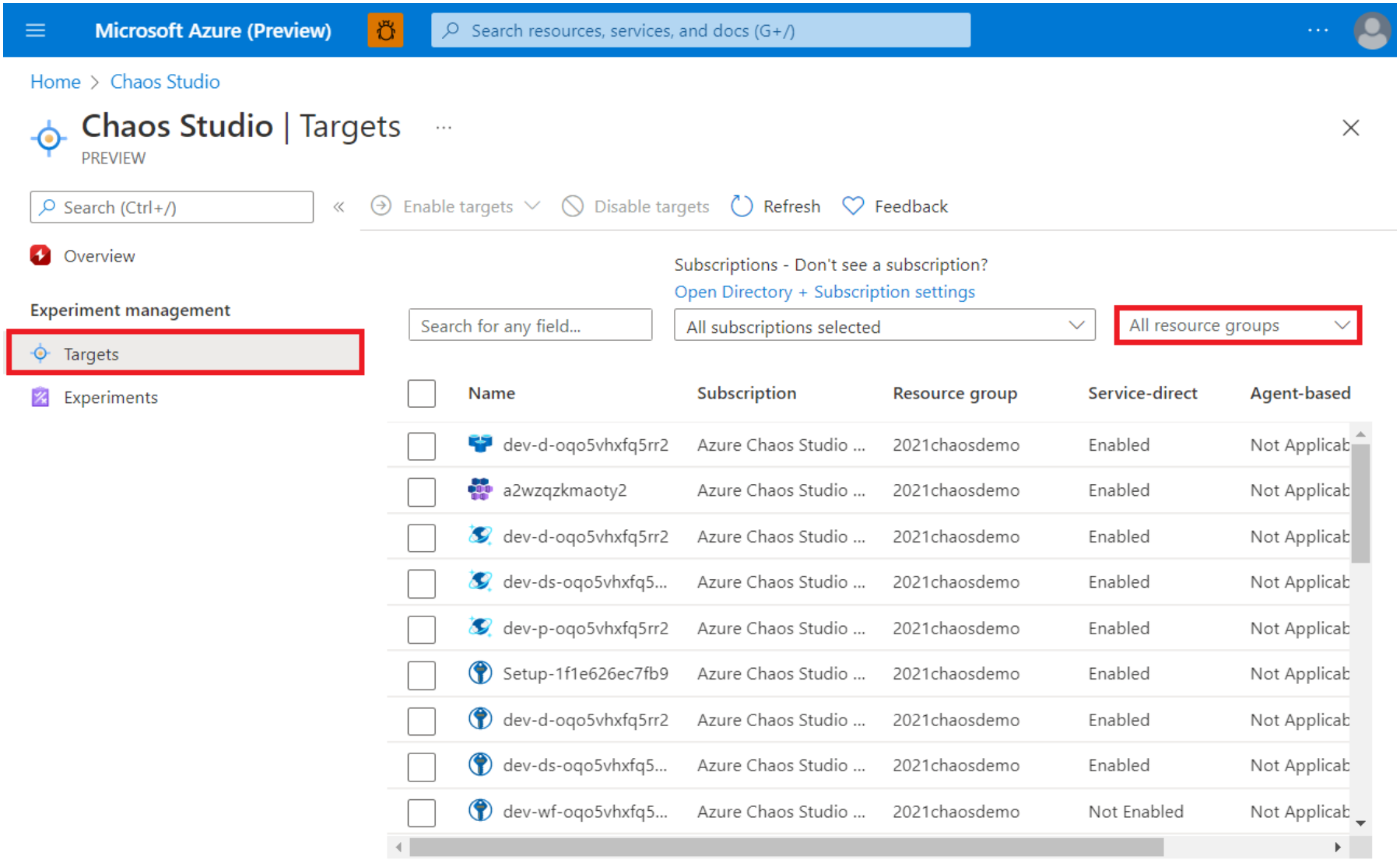
A hypothesis for this scenario might look like: “If a single availability zone goes down, availability should remain at **% with** % tolerance. I expect to find these results by analyzing __.” These variables, or the hypothesis altogether, may differ based on your environment.

Using this example, a potential hypothesis may be: “If a single availability zone goes down, availability should remain at 99% with 1% tolerance. I expect to find these results by analyzing my application’s availability monitoring service against my defined SLIs and SLOs.”

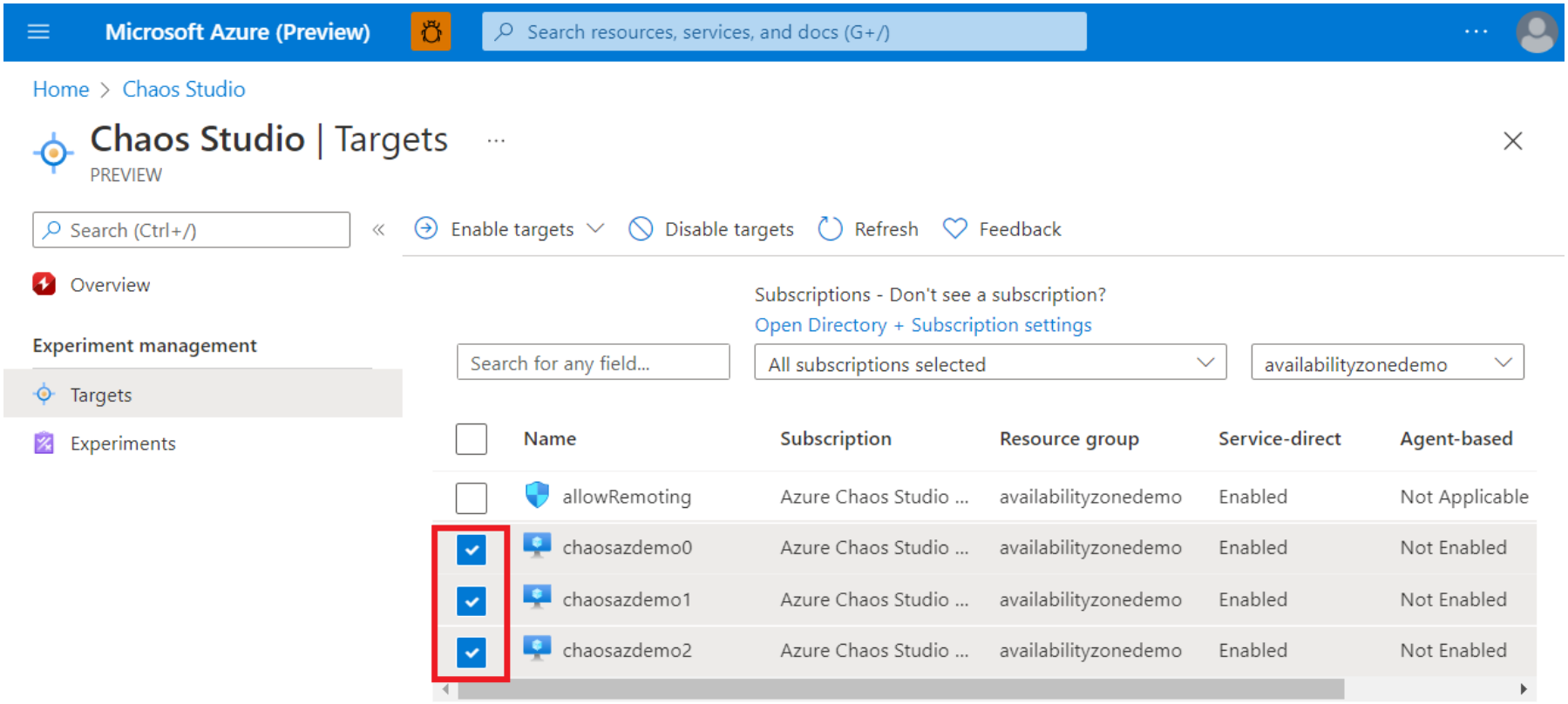
Onboard the resources

1. Open **Azure Portal**.
2. Search for **Chaos Studio** in the search bar.
3. In Chaos Studio, select **Targets**, then select the resource group you want the experiment to use from the **All resource groups** menu.

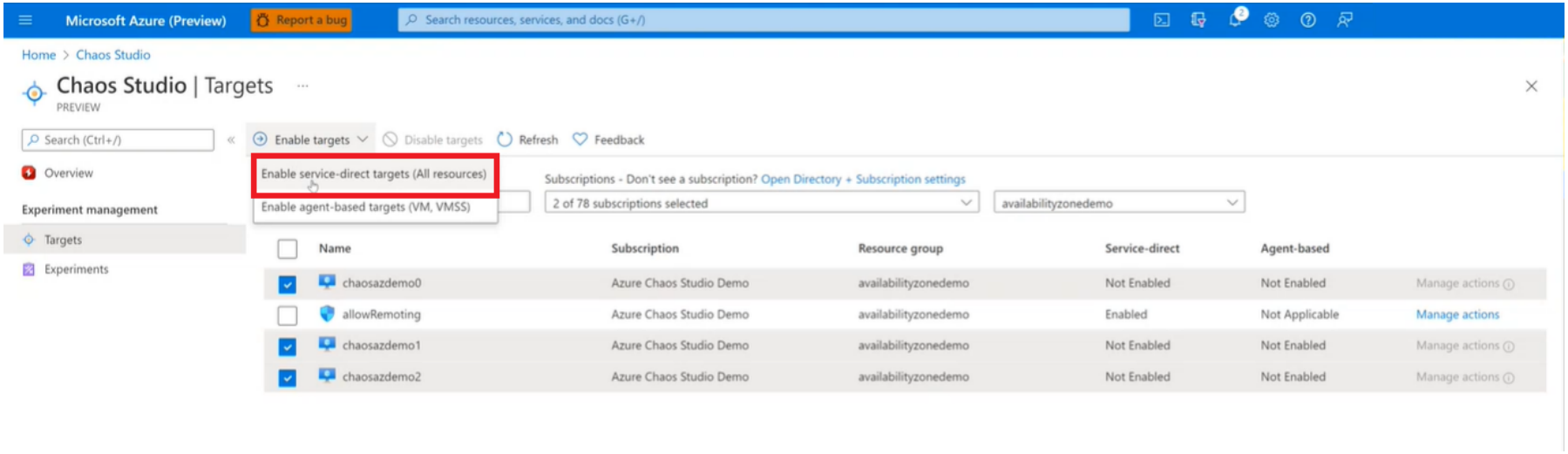
   Note: You will only see resources in regions where Chaos Studio is available (https://azure.microsoft.com/global-infrastructure/services/?products=chaos-studio).



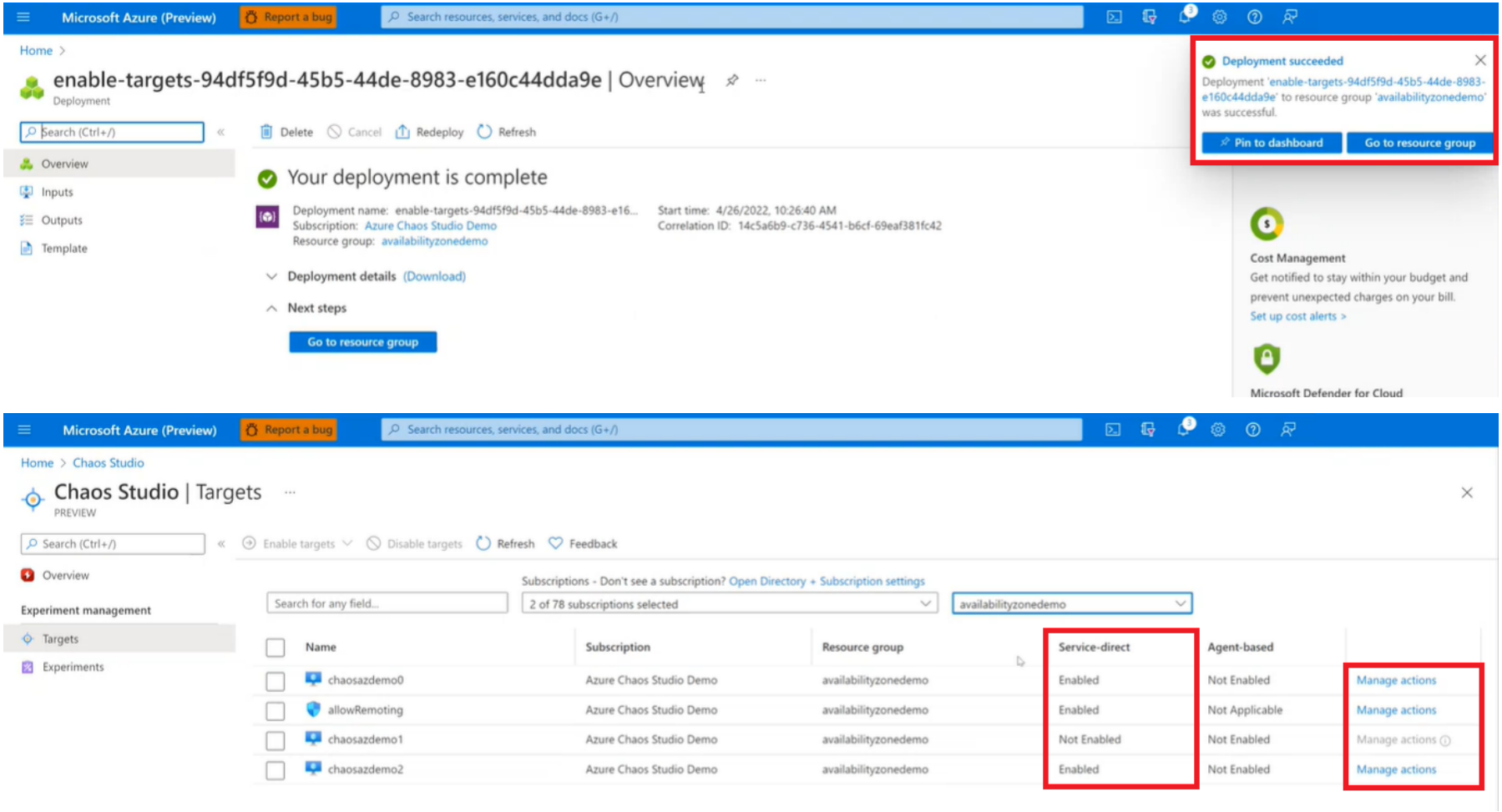
4. You will see a list of the resource group's contents. Select all your application's virtual machines that you wish to use when running the experiment.



5. Select **Enable targets**, then select **Enable service-direct targets (All resources)** from the dropdown menu.

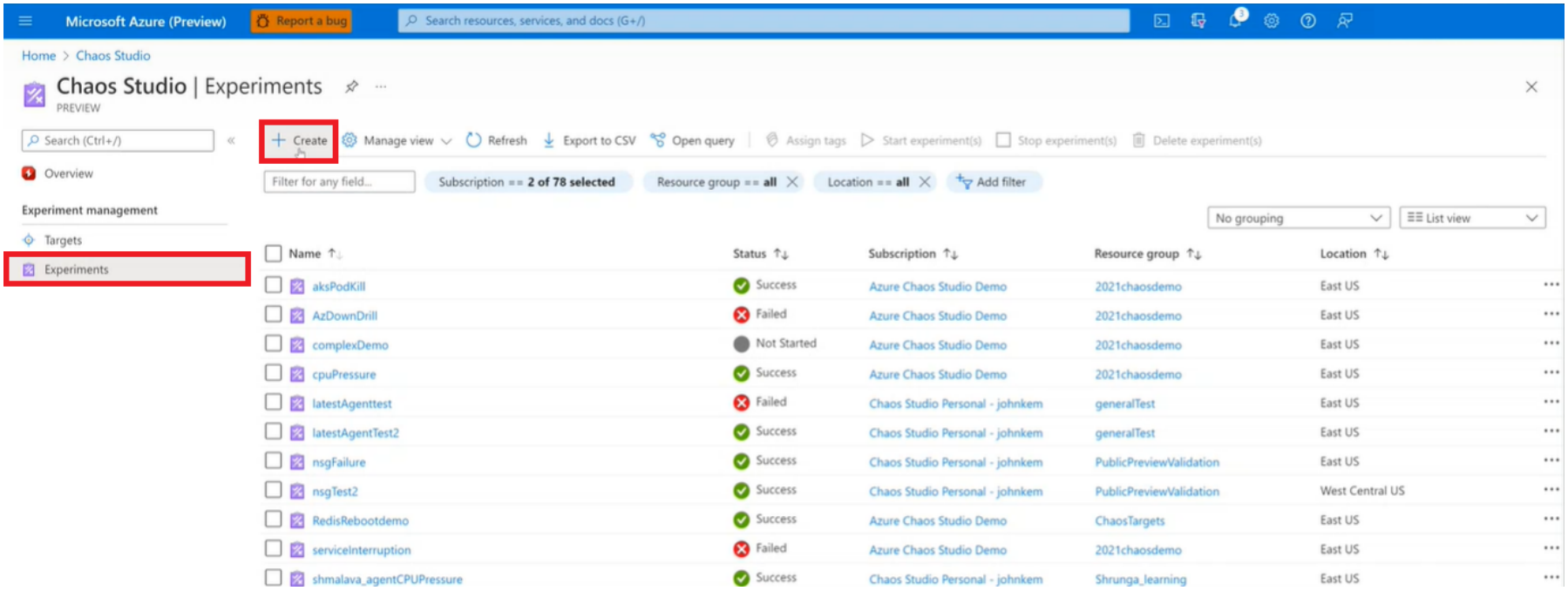


6. After a few minutes, you will see a **Deployment succeeded** notification indicating the targets were enabled successfully. In the **Targets** window, the virtual machines now display **Enabled** under the **Service-direct** heading and display active **Manage actions** links on the right.

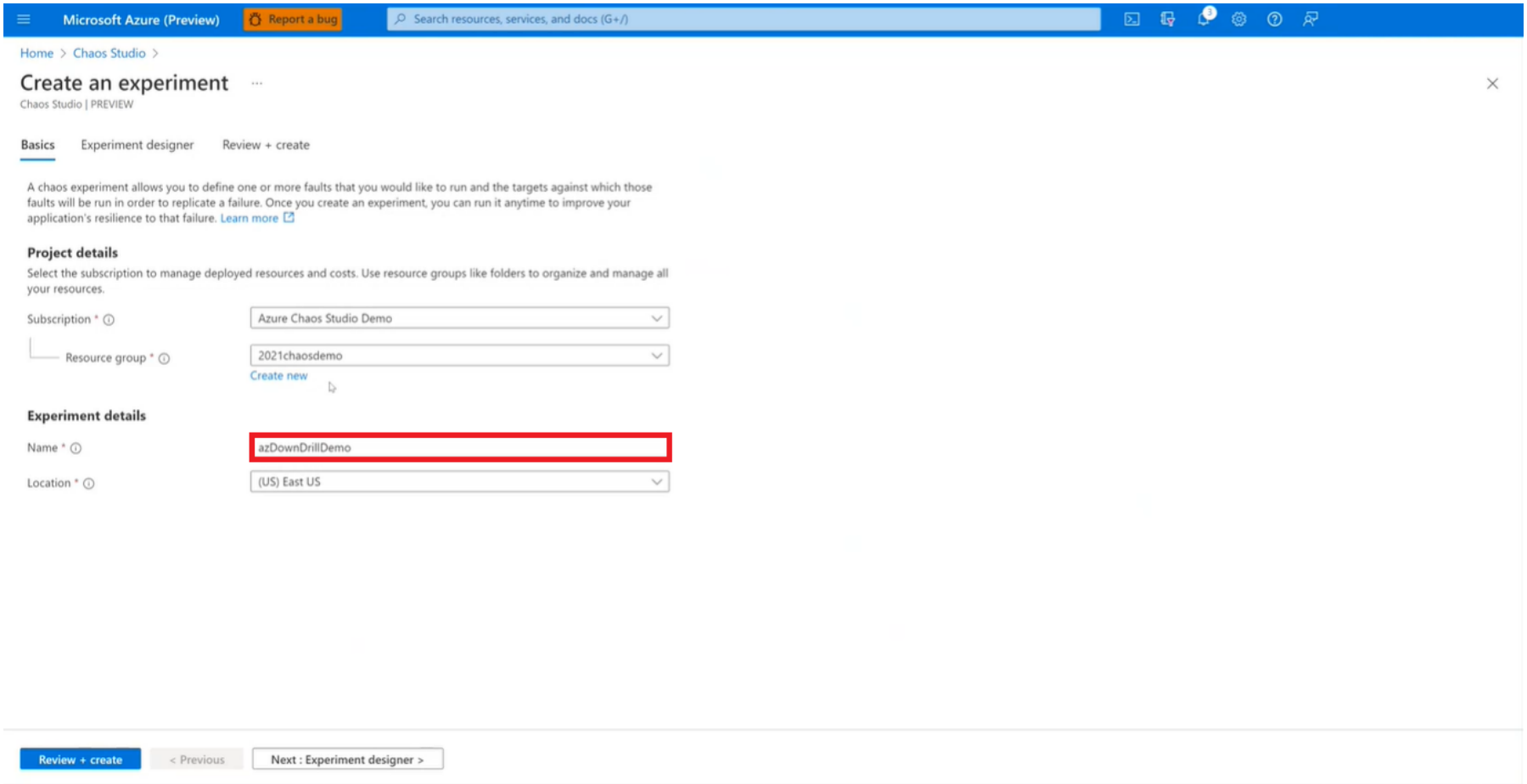


Create the experiment

1. Select **Experiments**, then select **Create**.

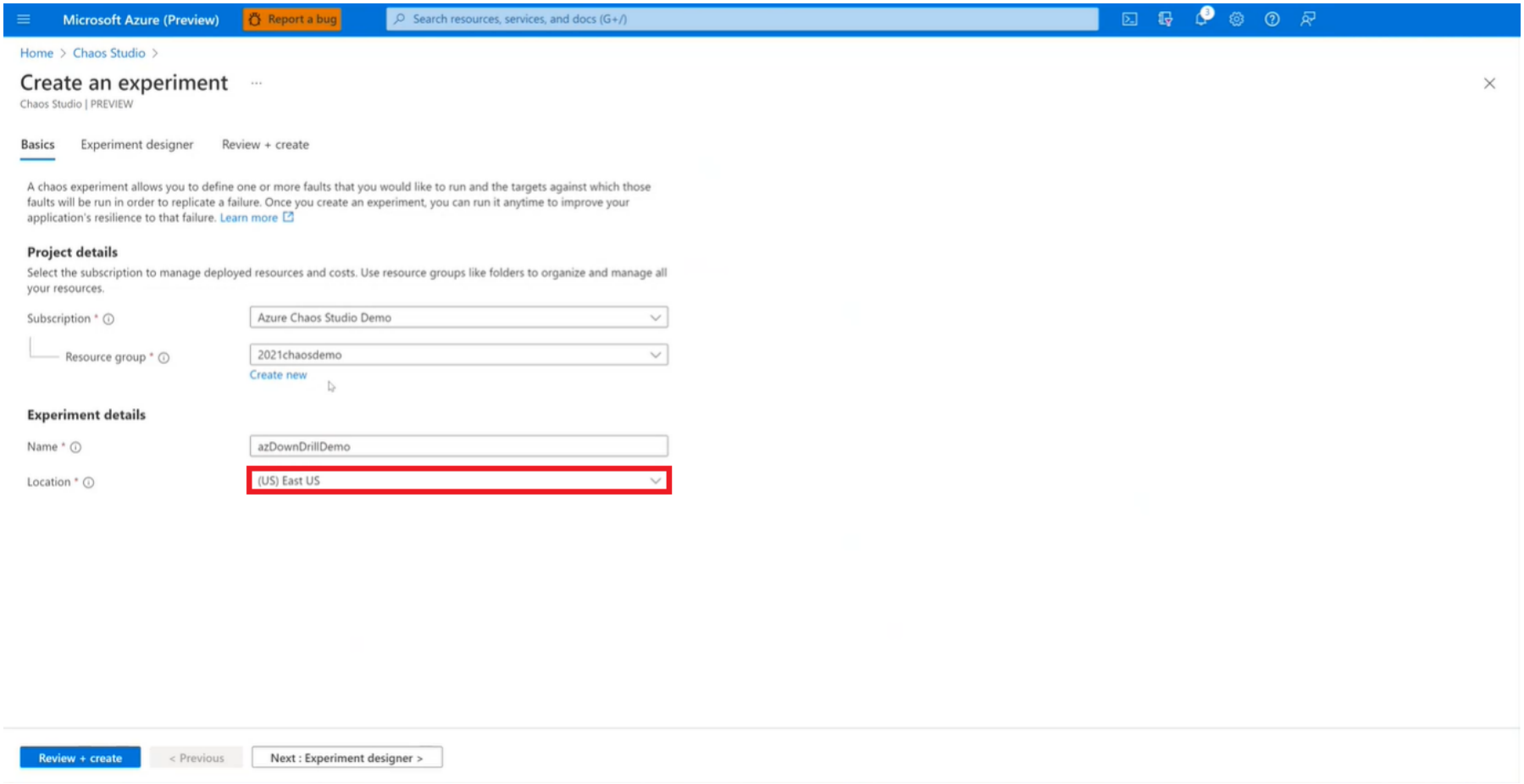


2. You will see the **Create an experiment** screen. In the **Name field**, enter a descriptive name for your experiment.

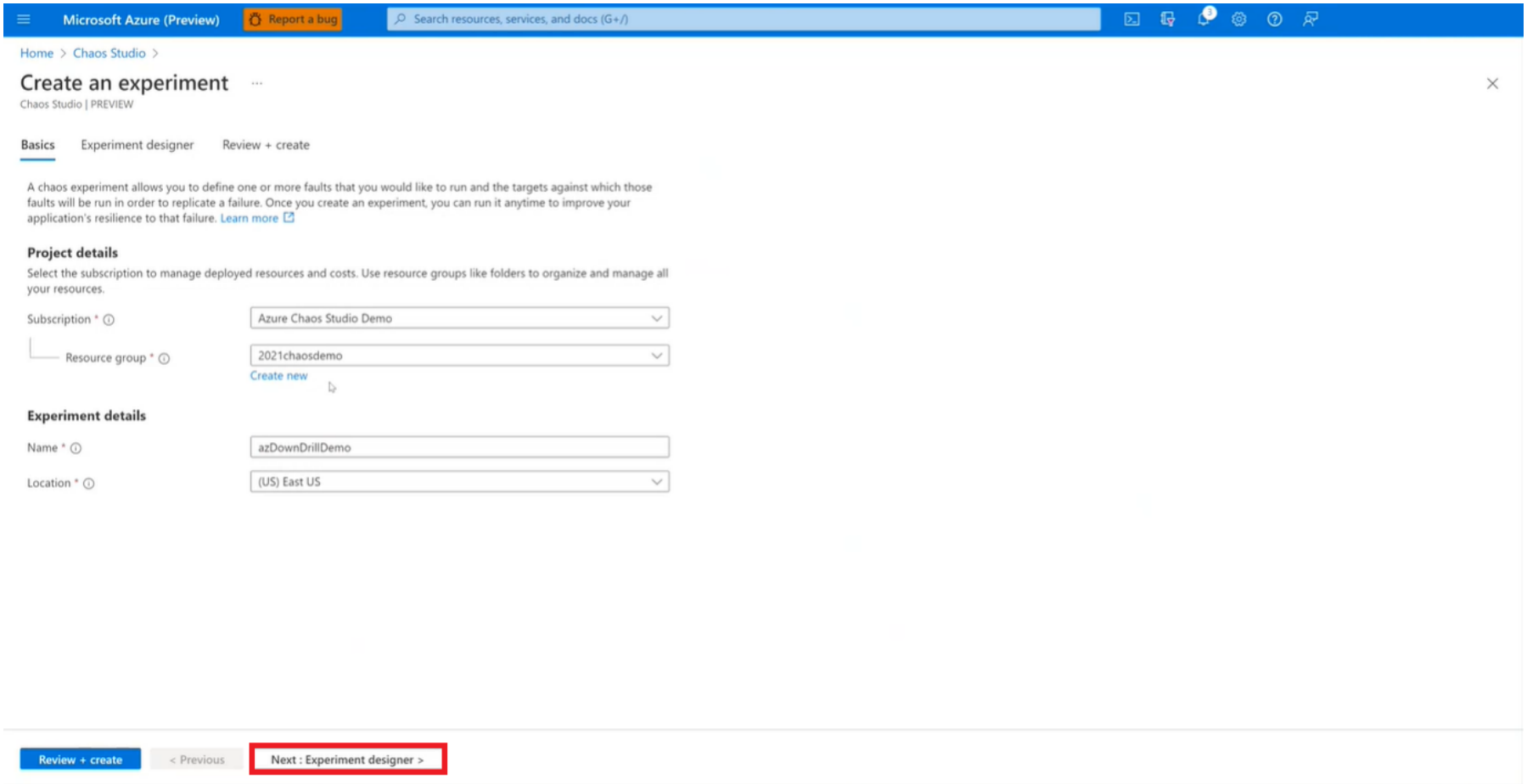


3. Select a region from the **Location** dropdown.

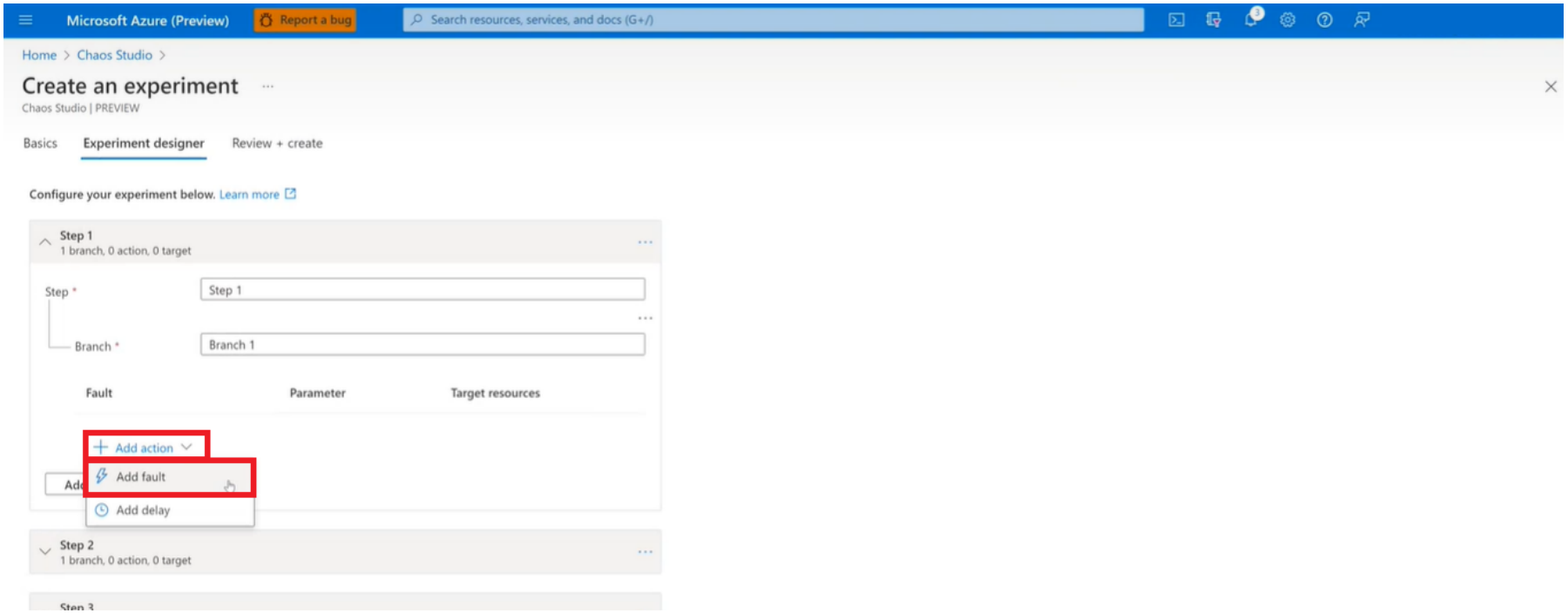
   Note: Chaos Studio must be available in the region you select. Refer to the Products available by region documentation (https://azure.microsoft.com/global-infrastructure/services/?products=chaos-studio) for a list of applicable regions.



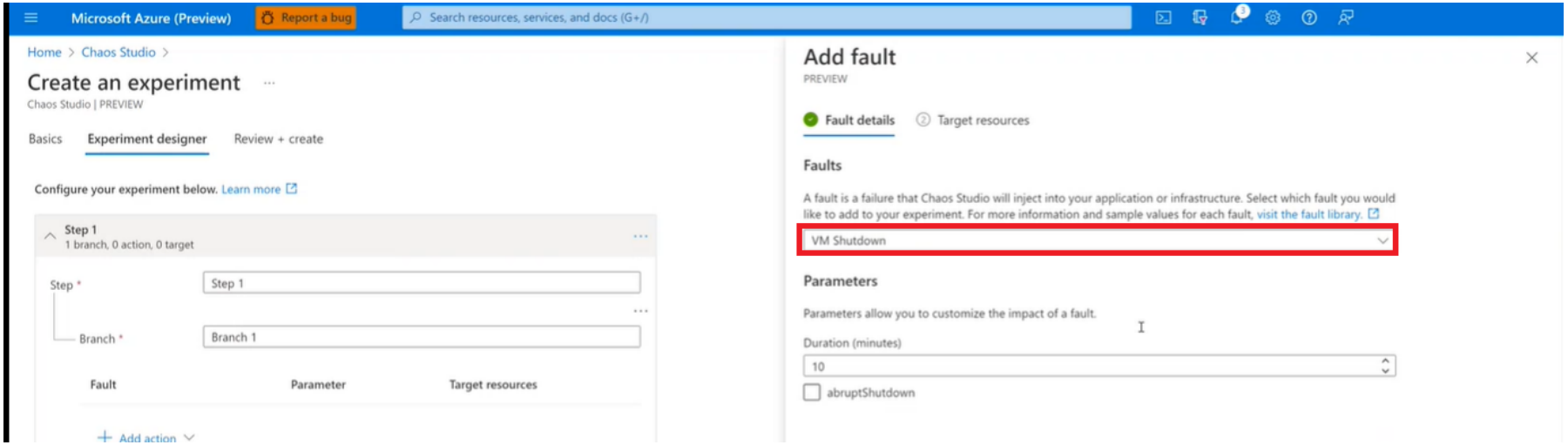
4. Select **Next: Experiment designer**.



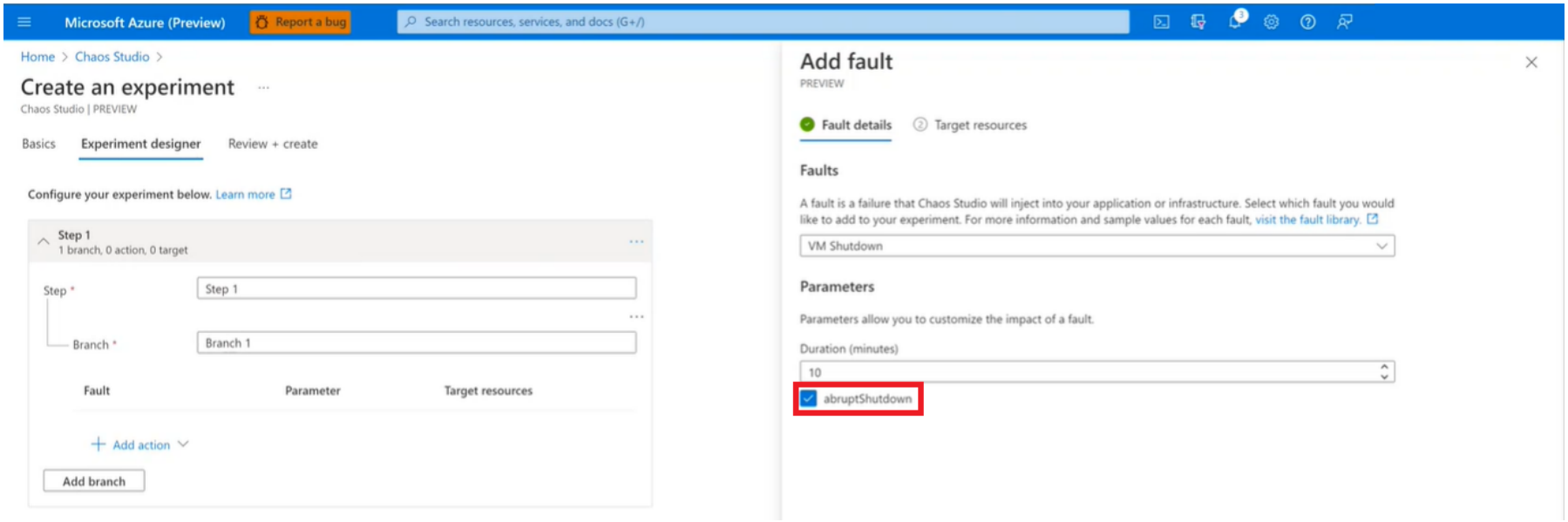
5. Select **Add action**, then select **Add fault** to add a fault to the step.



6. Select the **Select a fault** dropdown, then select **VM Shutdown** from the fault library.

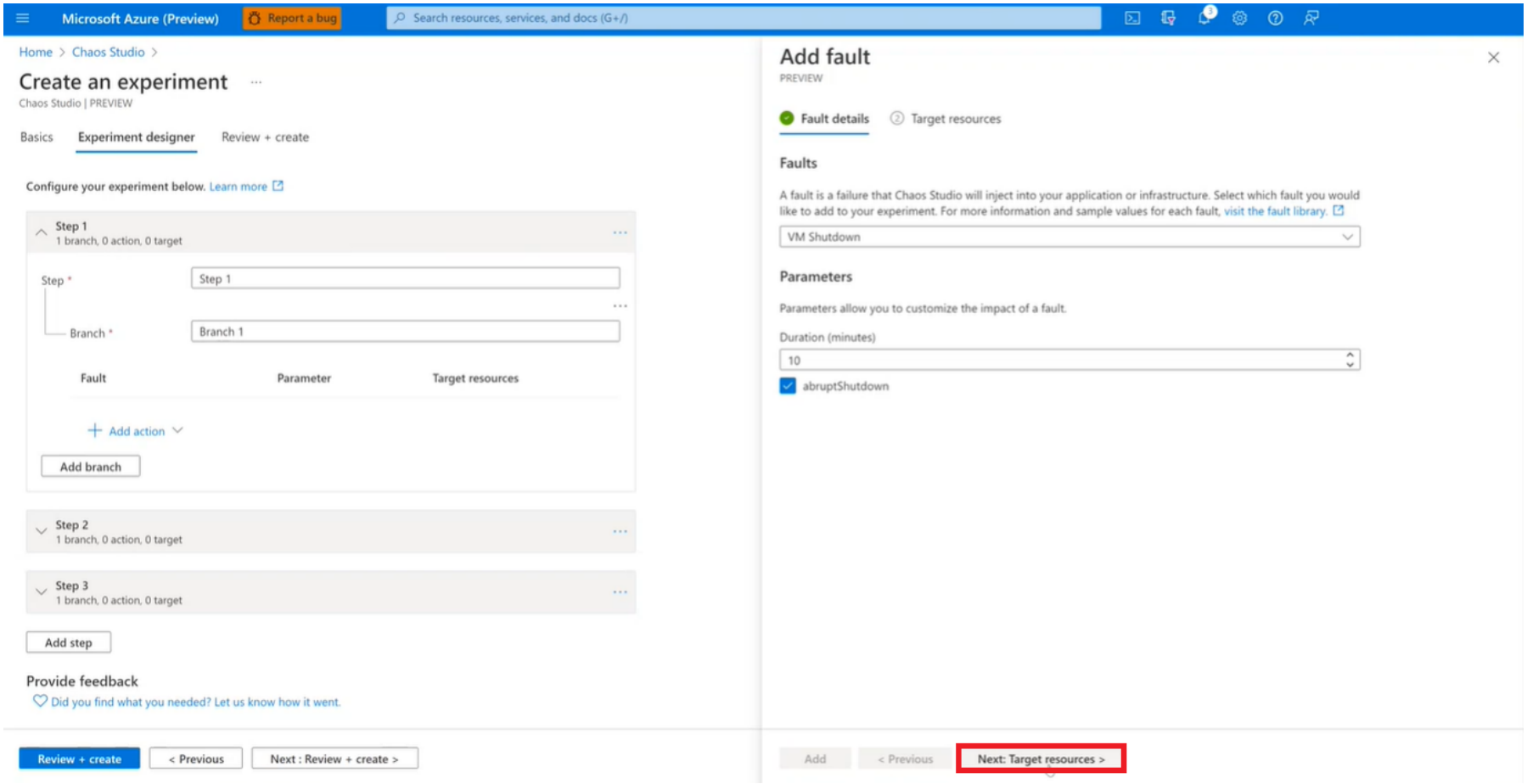


7. You will see a list of parameters specific to the VM Shutdown fault. Select **abruptShutdown**. The **abruptShutdown** parameter most closely replicates sudden zonal failure.



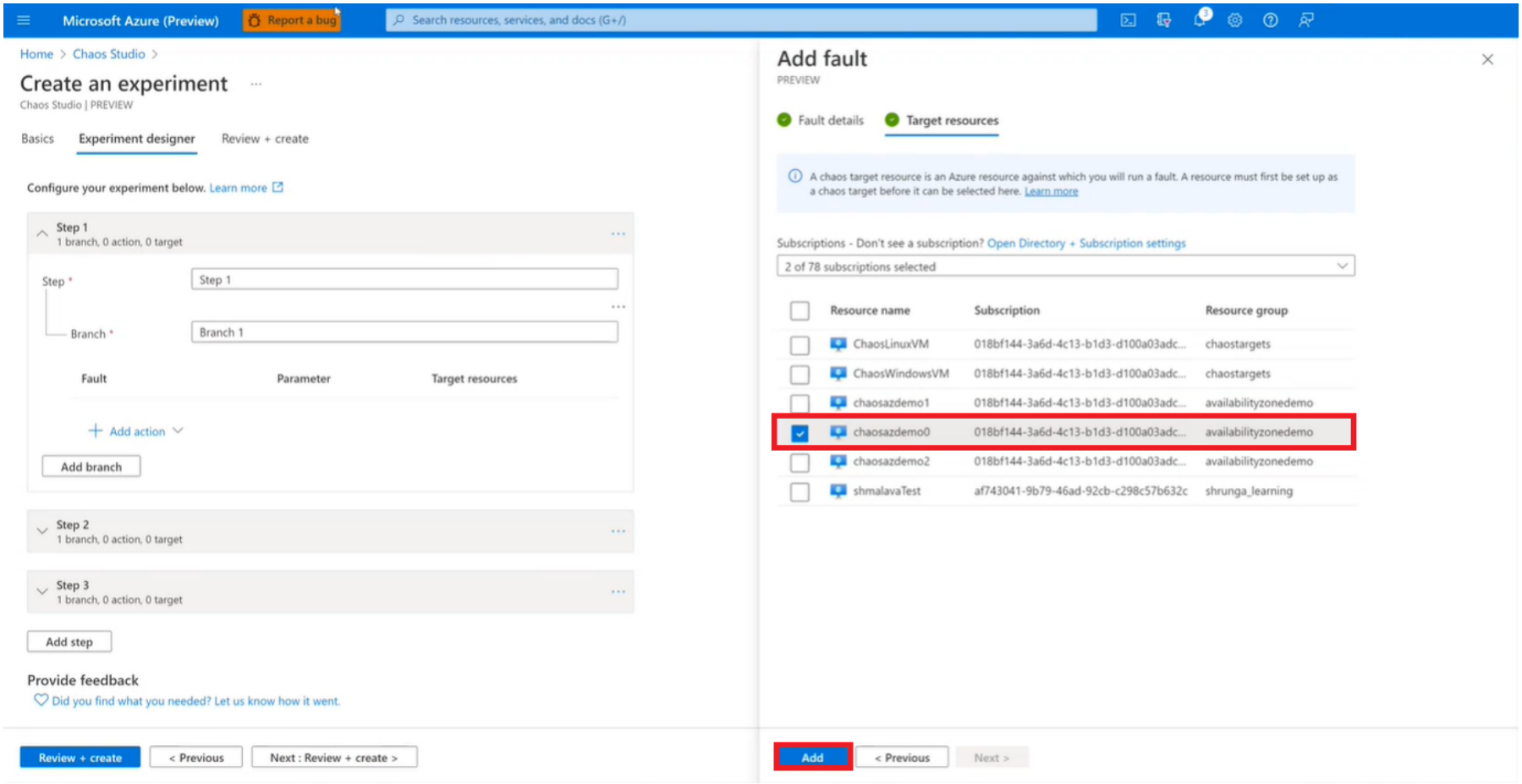
Note: The **Duration (minutes)** parameter is set to 10 minutes by default. Although not required, you can change this value to increase or decrease the experiment's runtime. 30 minutes is recommended to allow you to best observe the experiment's impact.

8. Select **Next: Target resources**.

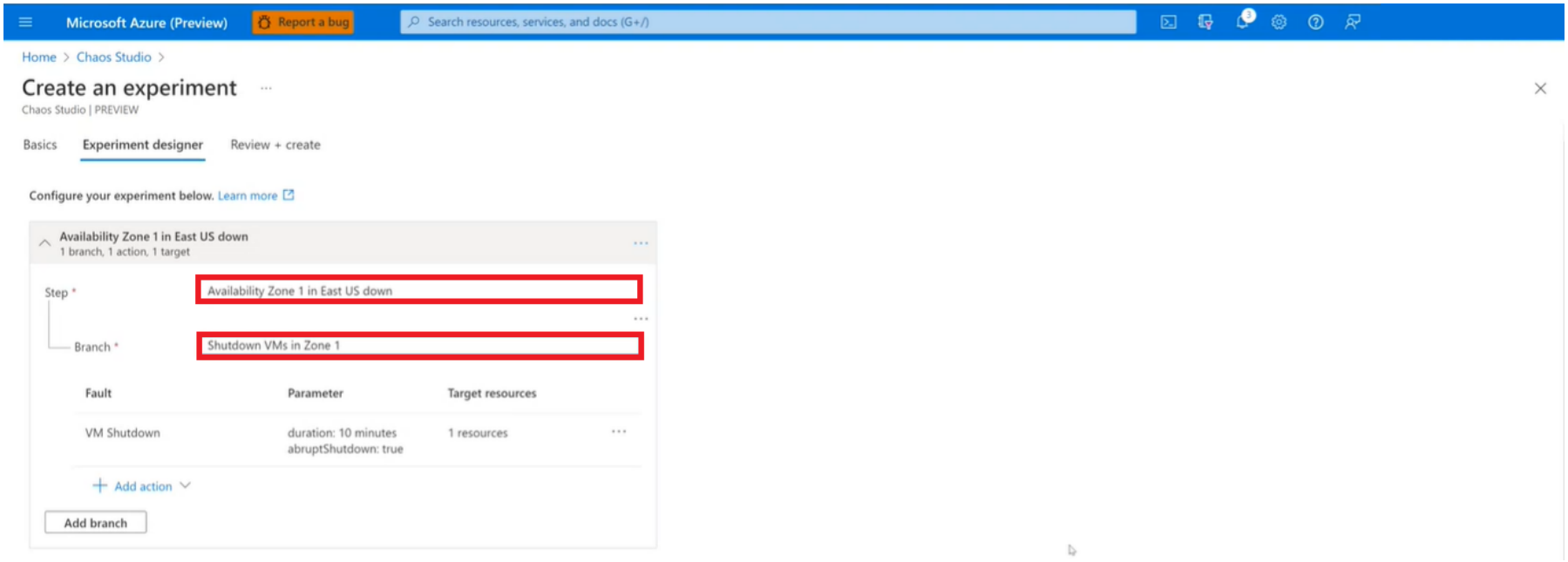


Note: This displays a list of virtual machines with service-direct faults enabled. Select the resource group's virtual machines to apply the experiment to. Each virtual machine represents a zone. Selecting only one virtual machine replicates a single zonal failure, but you can select more than one virtual machine if desired to replicate multiple zonal failures.

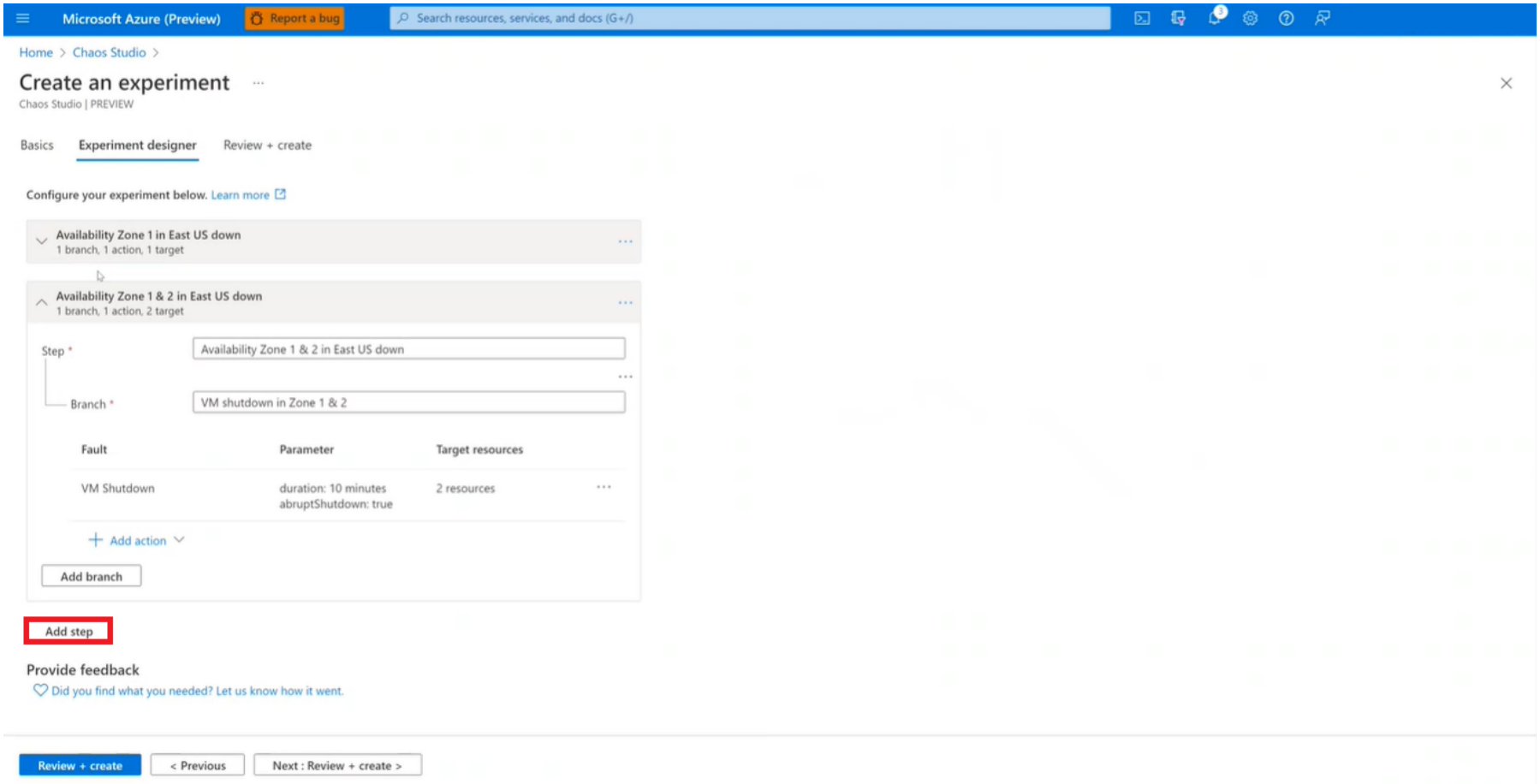
9. Select **Add**.



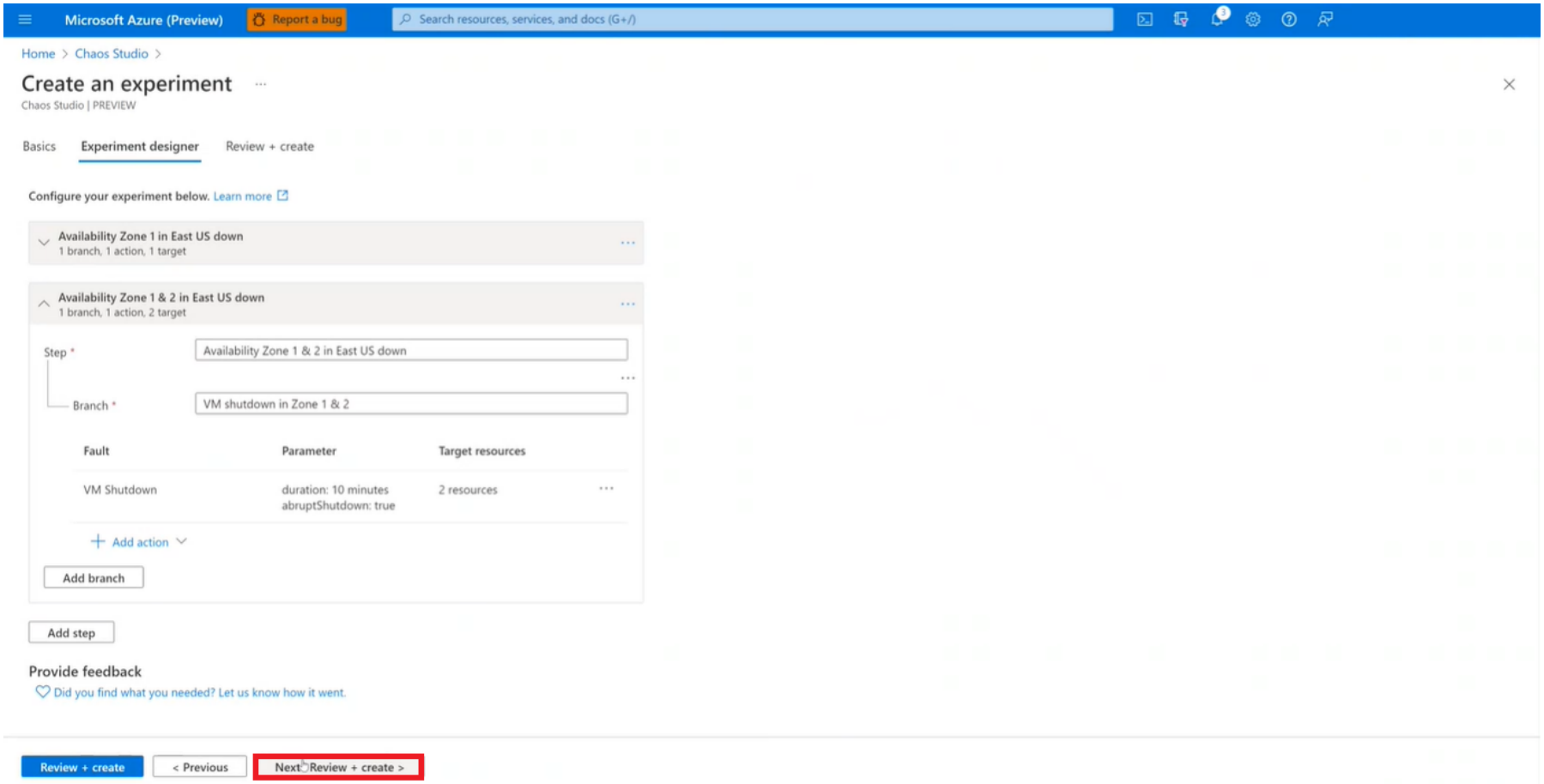
10. Enter descriptive names in the **Step** field and the **Branch** field.



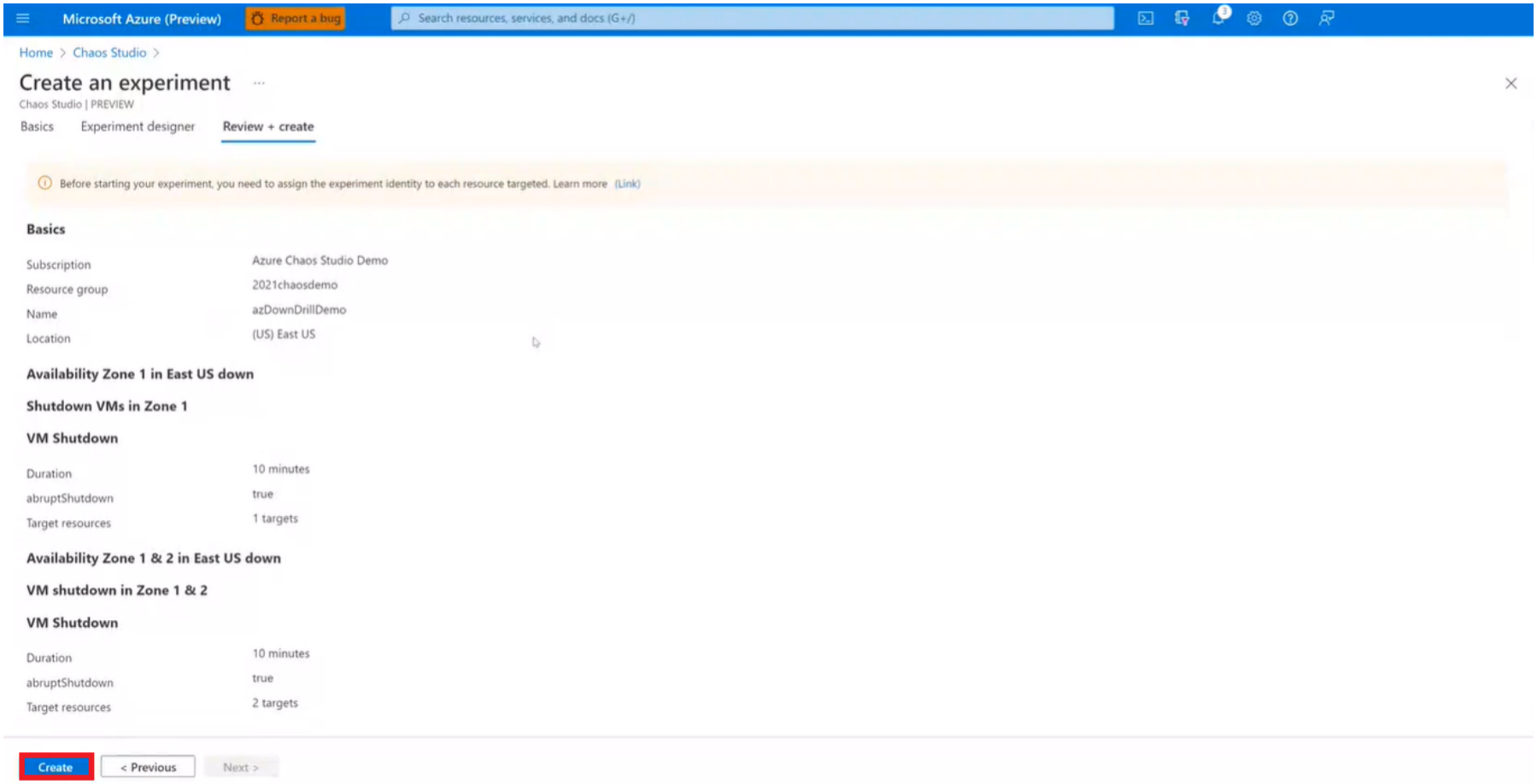
Note: If you are attempting to fail multiple availability zones, you can add more steps to the experiment. Each step is run sequentially. Select the **Add step** button and repeat steps 5-11 of this section for each new step.



11. Once you have added all desired steps, select **Next: Review + create**.



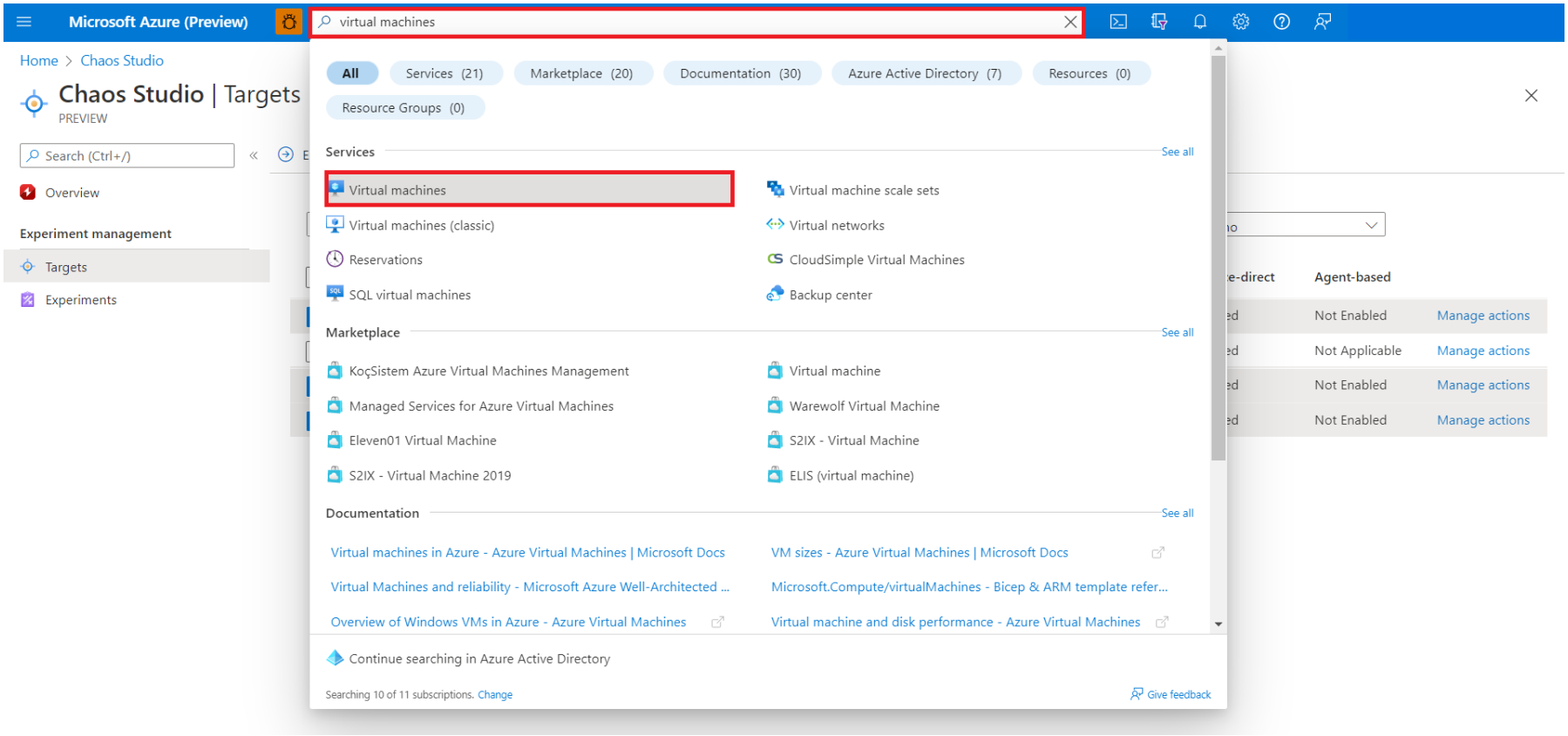
12. The **Review + create** screen appears. Review the experiment details. Then, when you are ready to proceed, select **Create**.



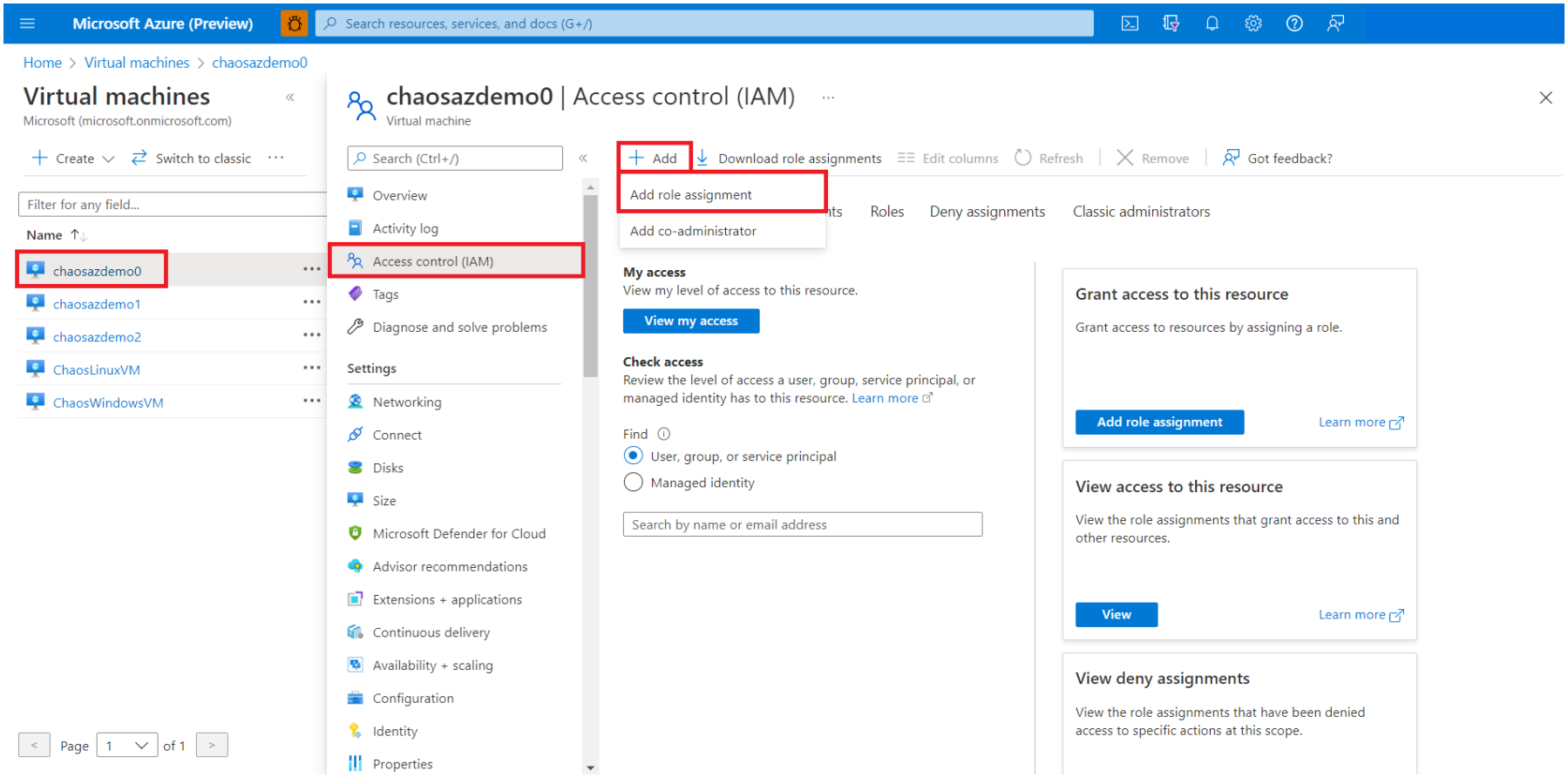
Assign an experiment identity to each targeted resource

Before starting the experiment, you need to assign an experiment identity to each targeted resource. The experiment will fail if an identity is not assigned.

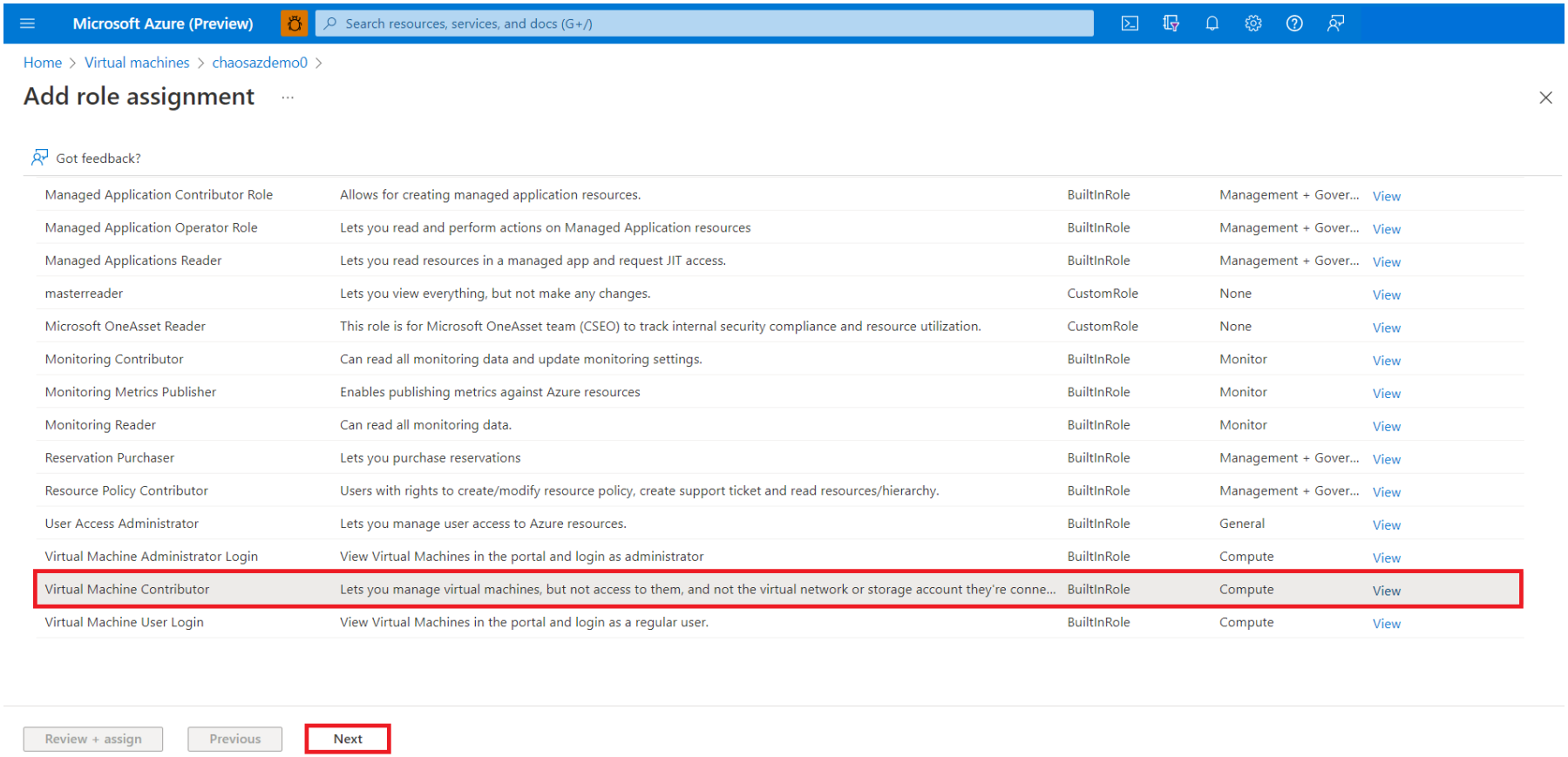
1. Search for **virtual machines** in the search bar and select **Virtual machines** from the menu.



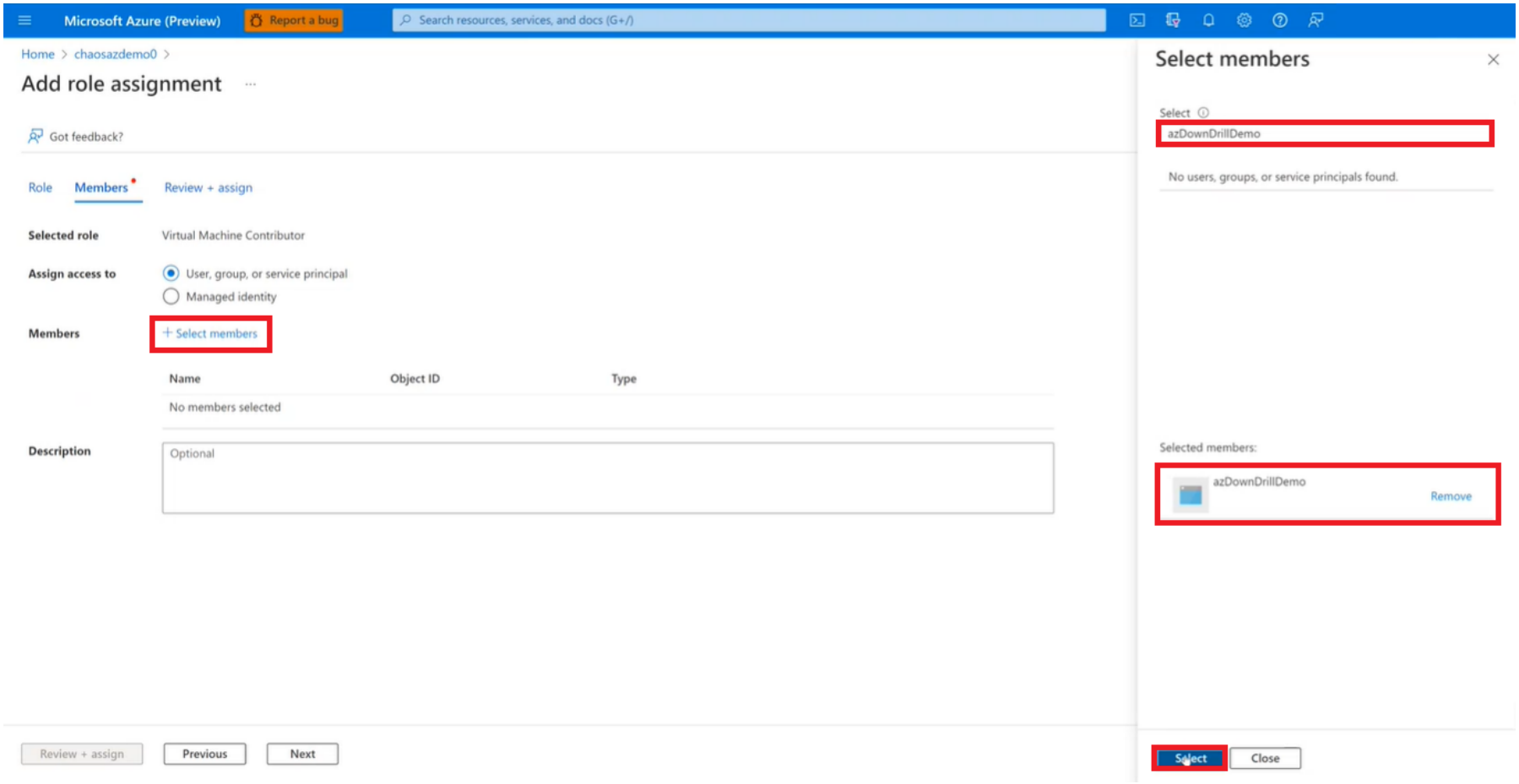
2. Select one of the resources targeted by any step of the created experiment to open its details panel. In this panel, select **Access control (IAM)**, select **Add**, then select **Add role assignment**.



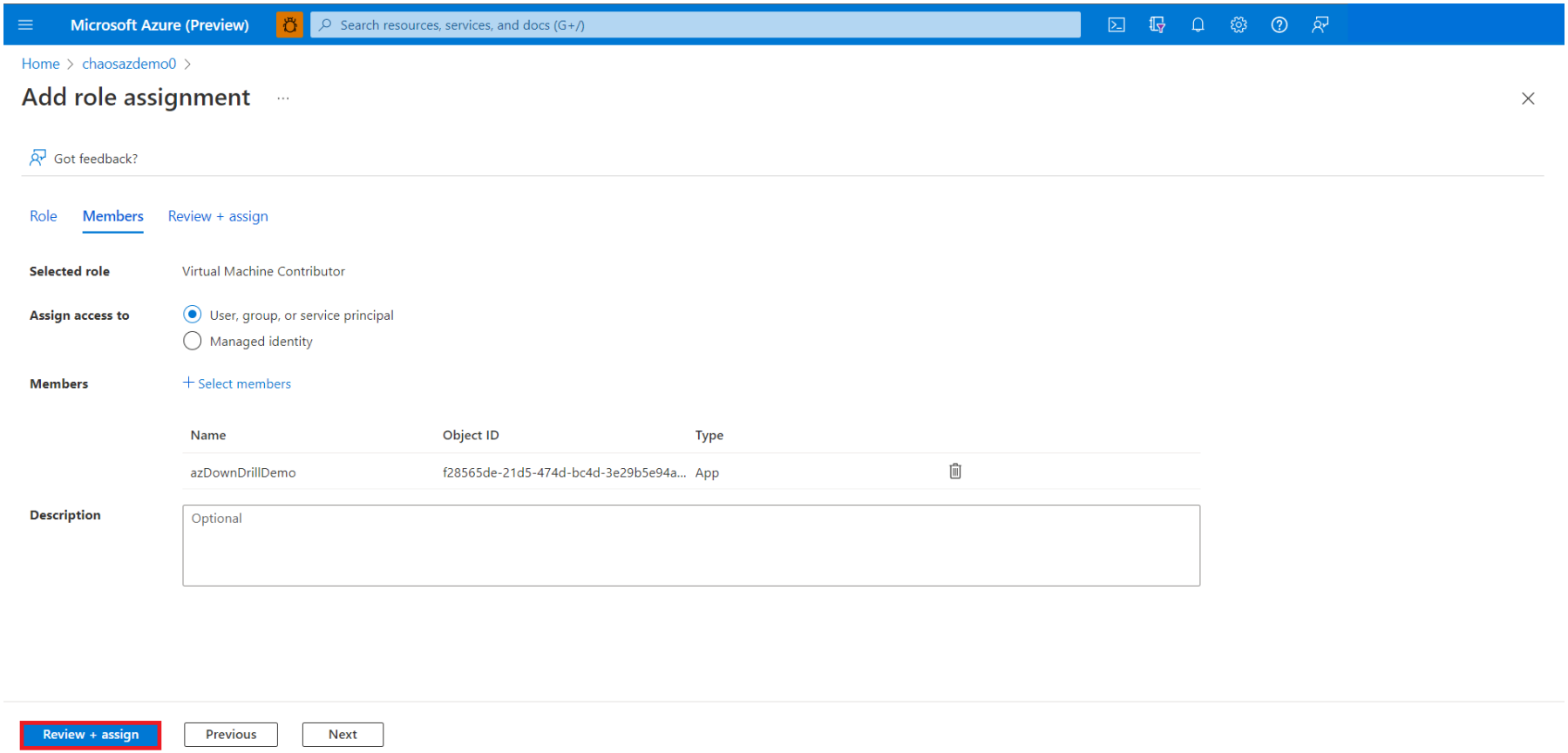
3. Select **Virtual Machine Contributor** from the list, highlighting it. Select **Next**.

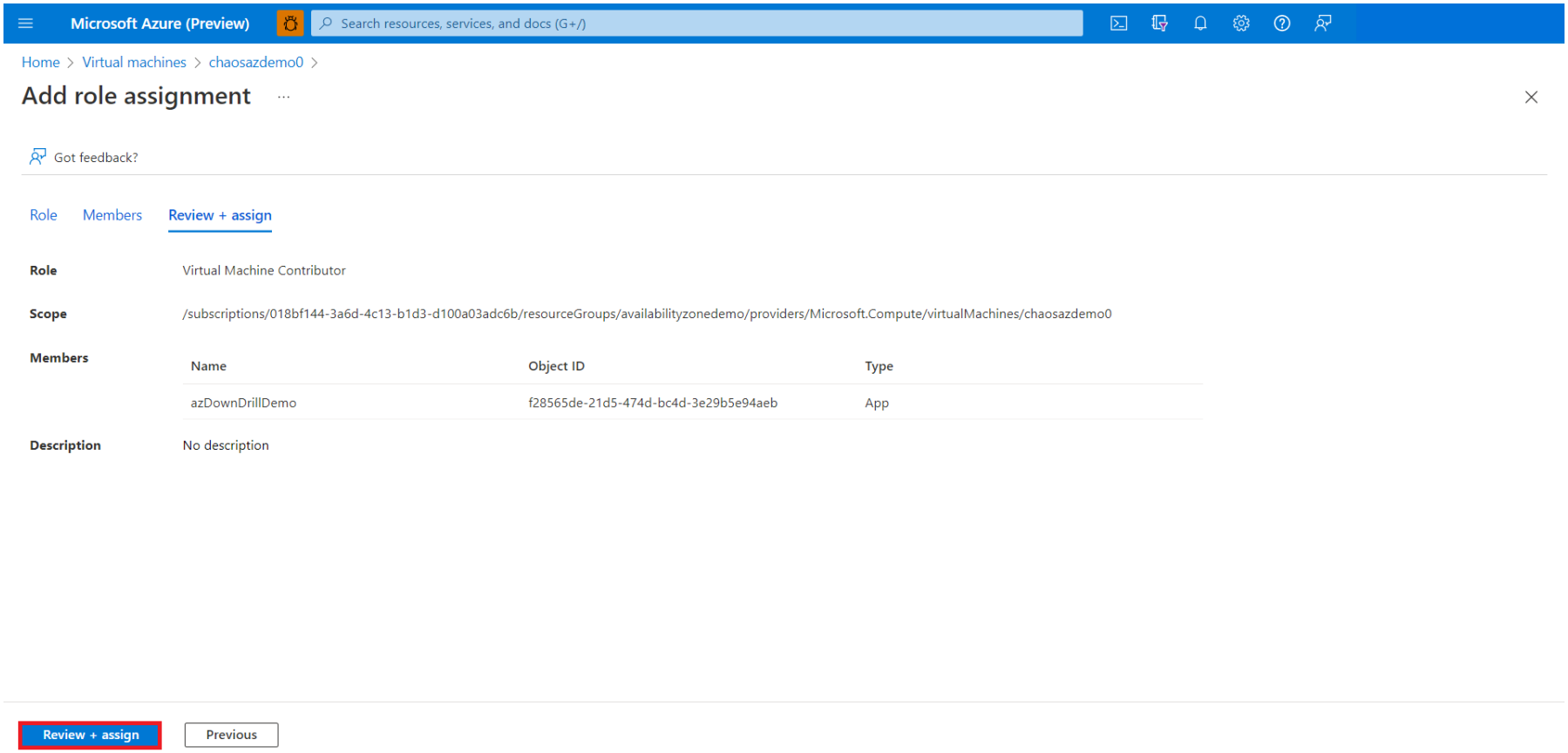


4. Select the **Select members** link. In the right sidebar, enter the name of the experiment you created in the **Select** search field. Select the experiment from the list. Once selected, the experiment will move to the **Selected members:** section. Select the **Select** button.



5. Select **Review + assign**, then select **Review + assign** again to assign the identity to the resource.

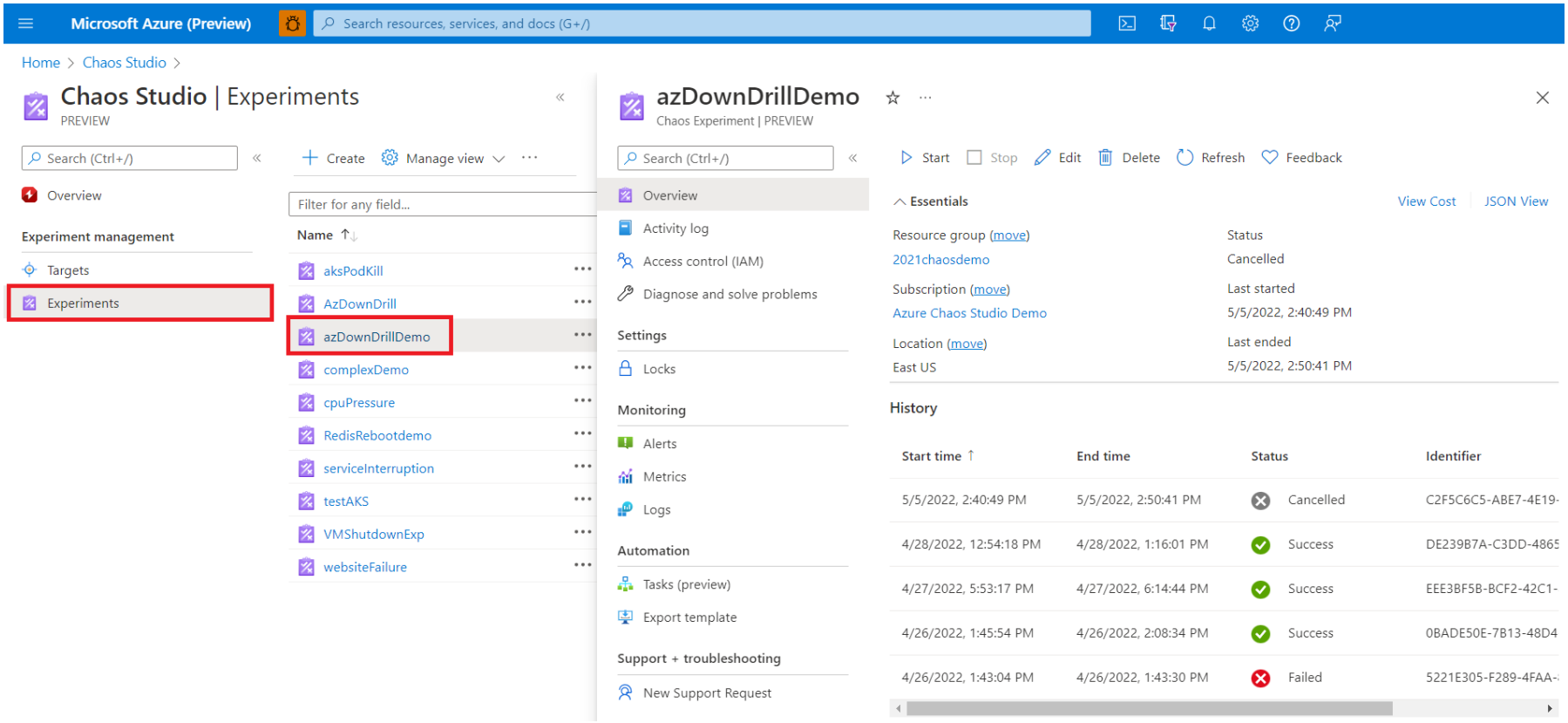

Microsoft Azure (Preview)
Search resources, services, and docs (G+/)
Home > chaosazdemo0 >
Add role assignment
Got feedback?
Role
Members
Review + assign
Selected role
Virtual Machine Contributor
Assign access to
User, group, or service principal
Managed identity
Members
+ Select members
Name
Object ID
Type
azDownDrillDemo
f28565de-21d5-474d-bc4d-3e29b5e94a...
App
Description
Optional
Review + assign
Previous
Next



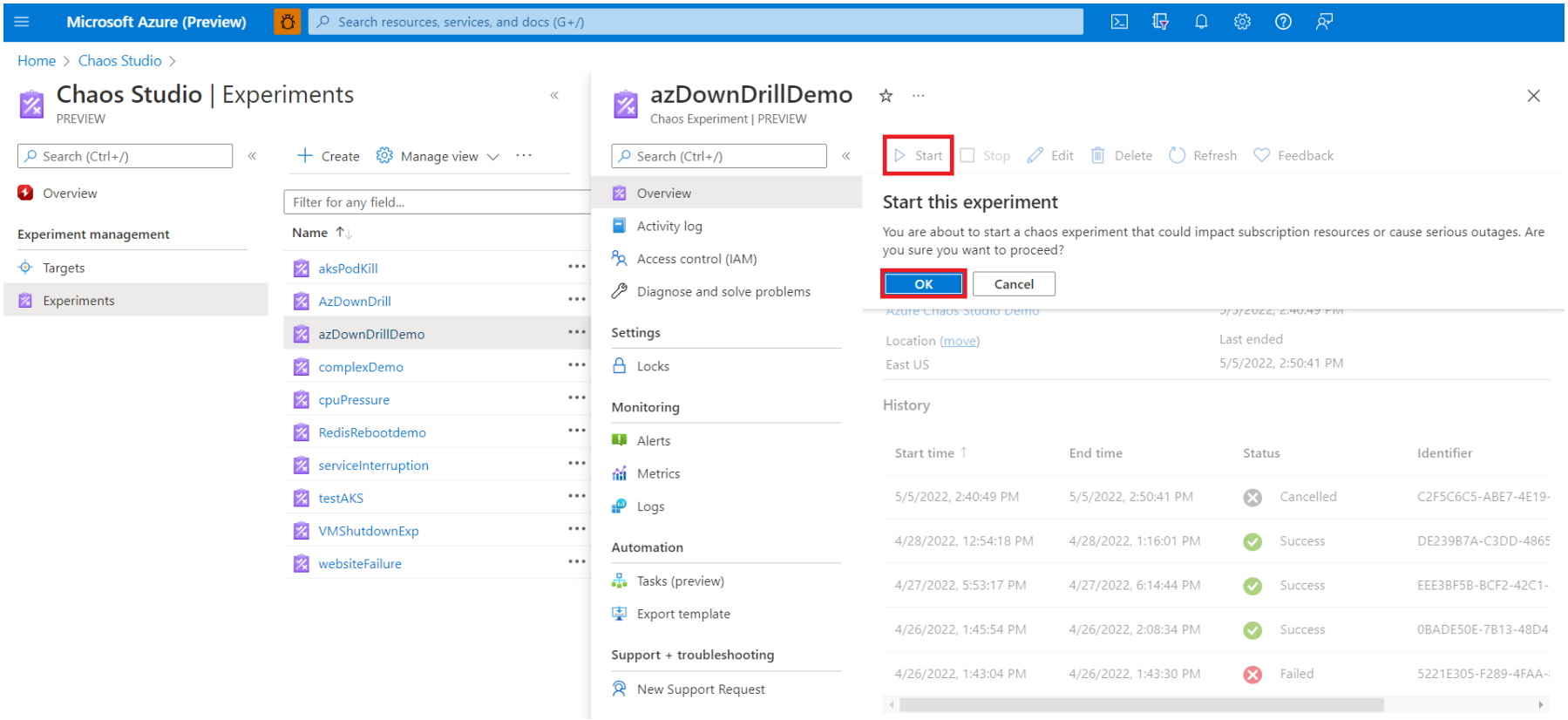
6. Repeat steps 1-6 of this section for each resource targeted in the experiment. If the experiment only targeted one resource, continue to the next section.

Run the experiment

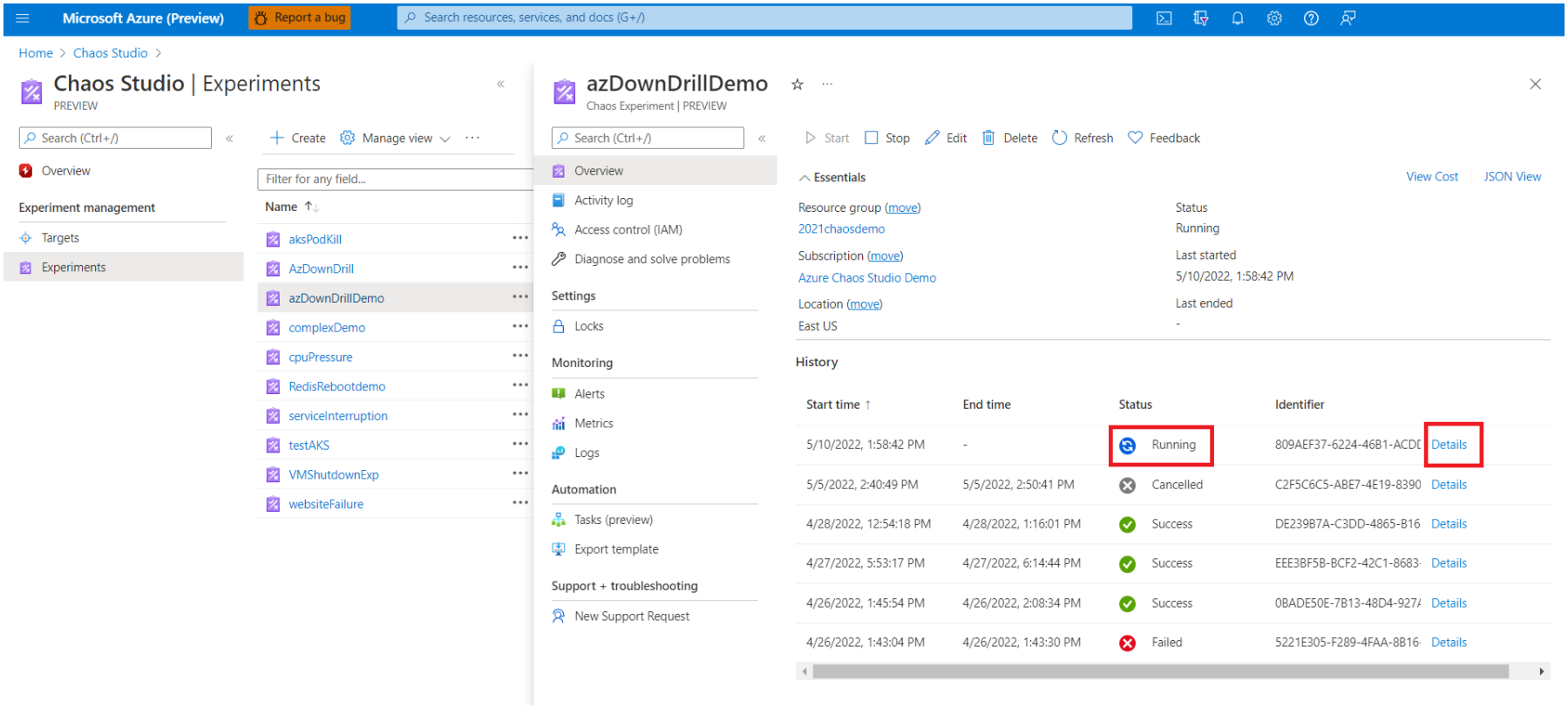
1. From the main Chaos Studio page, select **Experiments**, then select the name of your experiment. You will see the experiment details panel.



2. Select **Start**, then select **OK**.



3. Once the status changes to **Running**, the targeted virtual machines are successfully taken offline for the duration of the experiment. Select **Details** for real-time information on each branch and fault in the experiment.



Assess the hypothesis

Compare the results of the experiment against your hypothesis. Analyze any relevant metrics. Do the results align with your expectations?

For instance, if your hypothesis addresses availability testing, analyze your health model in Geneva for the duration of the Chaos experiment to see if there was any impact on availability. If there was an impact, analyze the returned logs and metrics from the experiment to understand why there was a dip in availability. Similarly, if you are testing to validate SLI alerts or to validate feedback on failures, analyze any feedback against the hypothesis to ensure the alerts are properly responding to failures.

If your results were unexpected, consider any reasons why, create a new hypothesis, implement any necessary changes, and repeat the experiment: “If a single availability zone goes down, availability should remain at **% with** % tolerance because **resilience improvement has been made. I expect to find these results by analyzing** .”

Overview

You have now learned about availability zones and observability metrics, how to formulate and evaluate an experiment hypothesis, and how to create an experiment in Azure Chaos Studio that tests the resiliency of an application during a zonal failure scenario.

Next steps

- Manage your experiment (https://docs.microsoft.com/azure/chaos-studio/chaos-studio-tutorial-service-direct-portal#:~:text=Manage%20your%20experiment)

Additional resources

- Troubleshoot issues with Azure Chaos Studio (https://docs.microsoft.com/azure/chaos-studio/troubleshooting)
- Chaos Studio fault and action library (https://docs.microsoft.com/azure/chaos-studio/chaos-studio-fault-library)