Chaos Engineering Scenario: Leverage Chaos Studio to test the impact of a DNS outage on an application dependency

Introduction

You can use a chaos experiment to verify that your application is resilient to Domain Name System (DNS) outages by causing those outages in a controlled environment. By following this guide, you will simulate a DNS outage on an application dependency for an application running on Windows virtual machine(s) using a chaos experiment and Azure Chaos Studio. Ultimately, this helps you defend against service interruption.

Prerequisites

Before proceeding, you should understand:

- Your service's or application's dependency on Domain Name Systems (DNS) (https://support.microsoft.com/topic/description-of-domain-name-system-dns-d7476f12-818e-1db7-aa7b-7066fb5e382a)
- 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 Windows virtual machines in Azure Canary regions and follows mandatory safe deployment practices (https://eng.ms/docs/quality/zero-self-inflicted-sev1s/safedeploy). DNS testing with Chaos Studio is currently only available using Windows virtual machines.
- A user-assigned managed identity. (https://docs.microsoft.com/azure/active-directory/managed-identities-azure-resources/how-manage-user-assigned-managed-identities)

Scenario background

The DNS translates human-readable domain names like www.example.com (https://www.example.com/) into computer-readable IP addresses like 192.0.2.255. This is a key step in allowing users to easily access websites. During a DNS outage, the DNS server doesn’t connect a domain name to its corresponding IP address, leaving users without access.

Service level indicators (SLIs) are metrics such as availability, latency, throughput, and error rate. These metrics 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 to evaluate an application’s resilience to a DNS outage by analyzing any variance in the SLIs, which potentially violates the SLOs, when a DNS outage occurs.

Scenario goal

In this scenario, you will:

- Understand DNS, DNS outages, and DNS's relationship with application dependency, and understand how these elements can be 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 in the event of a DNS outage.
- 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 DNS and observability expectations. If there is a DNS outage and it causes a failure to resolve one or more of your application's dependencies, what do you expect to happen, and how do you expect to receive the results?

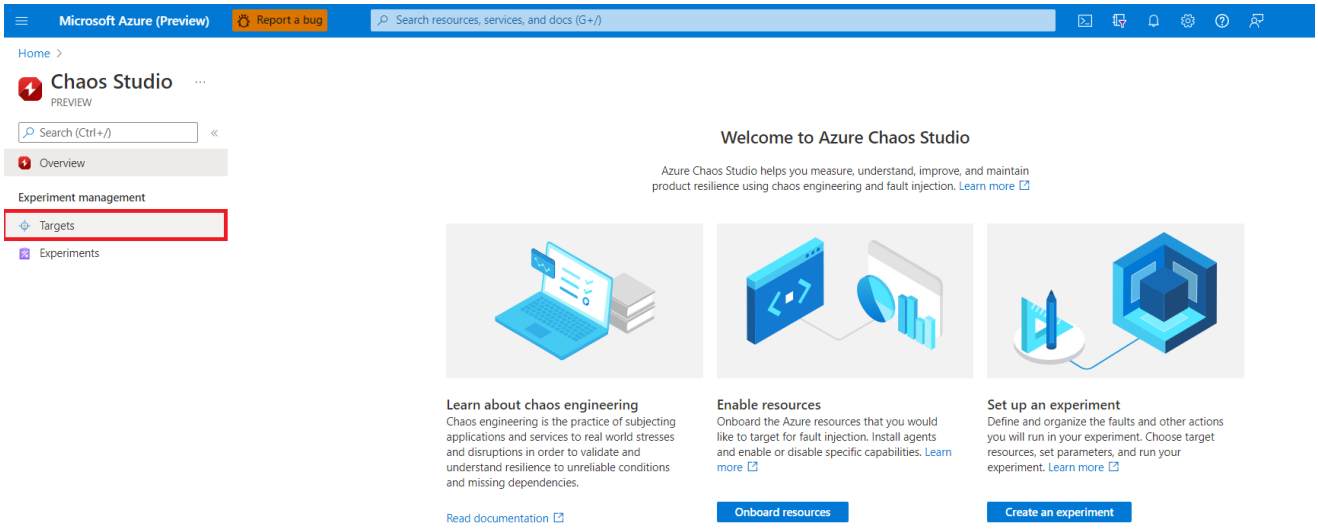
To create a hypothesis, ask questions relevant to the scenario. For example, what resilience measures are already in place to mitigate the impact of a DNS outage? Do these resilience measures work as expected? By running this experiment, what do you expect to happen given your specific application setup, 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: "In the event of a DNS outage, **ICM incidents were created, and the appropriate resiliency measures were activated. I expect to find the experiment results by analyzing** ." The hypothesis may differ based on your environment.

Using this example, a potential hypothesis may be: "In the event of a DNS outage, no ICM incidents were created, and the appropriate resiliency measures were activated. I expect to find the experiment results by analyzing my application's availability metrics against my defined SLIs and SLOs."

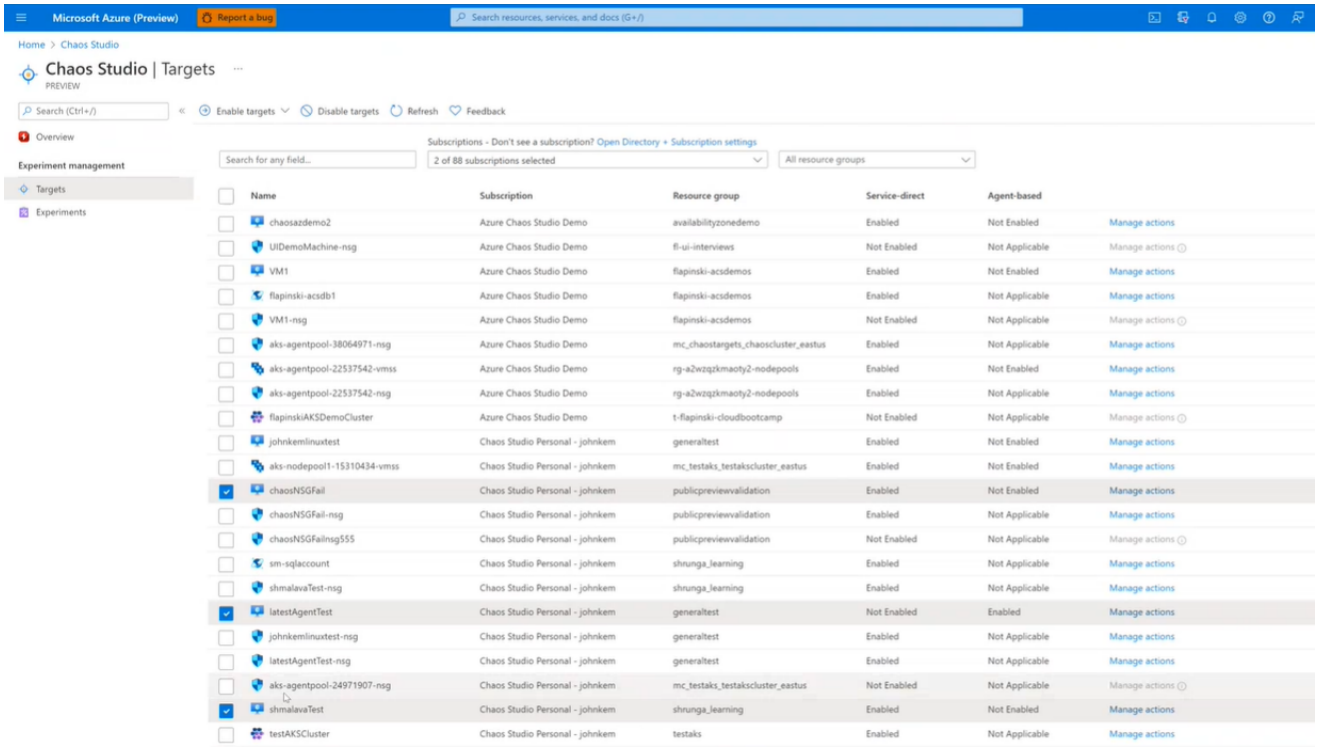
Onboard the resources

1. Open **Azure Portal**.
2. Search for **Chaos Studio** in the search bar.
3. In Chaos Studio, select **Targets**.

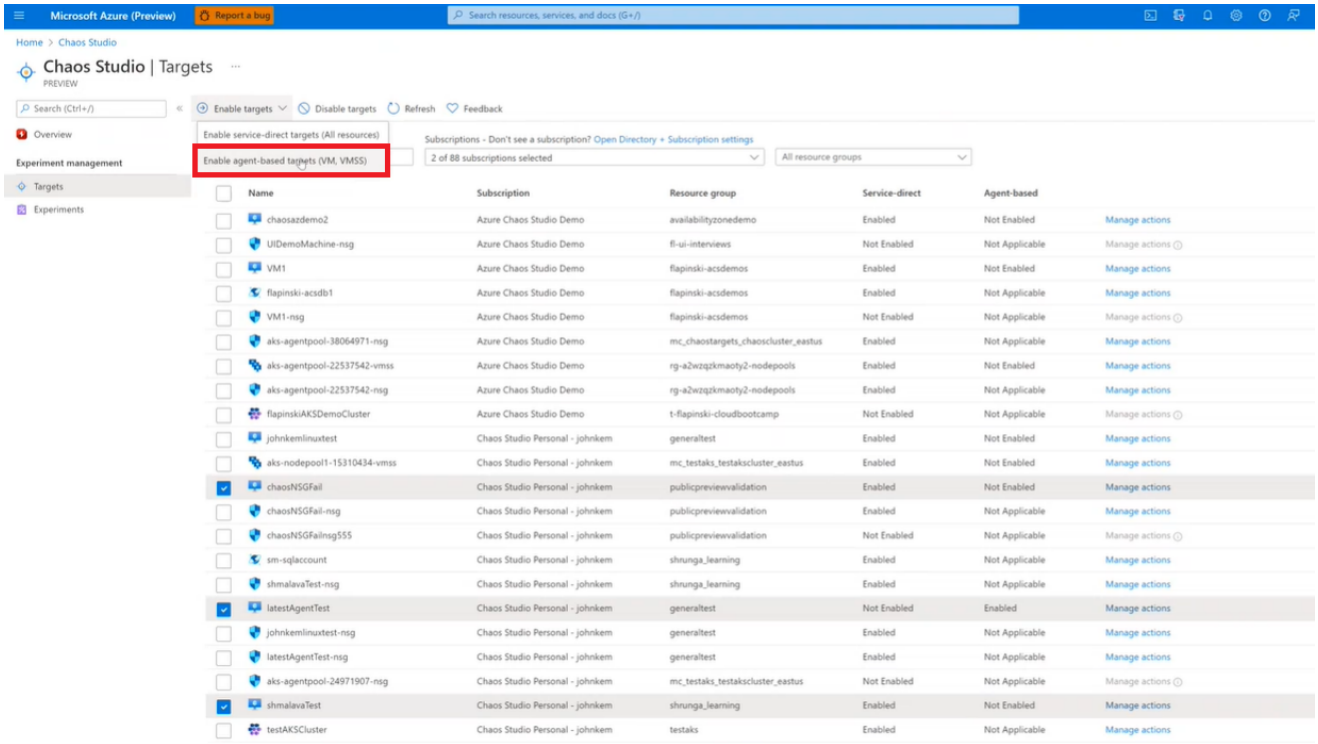


4. Select the virtual machine(s) or virtual machine scale set you wish to use when running the experiment.

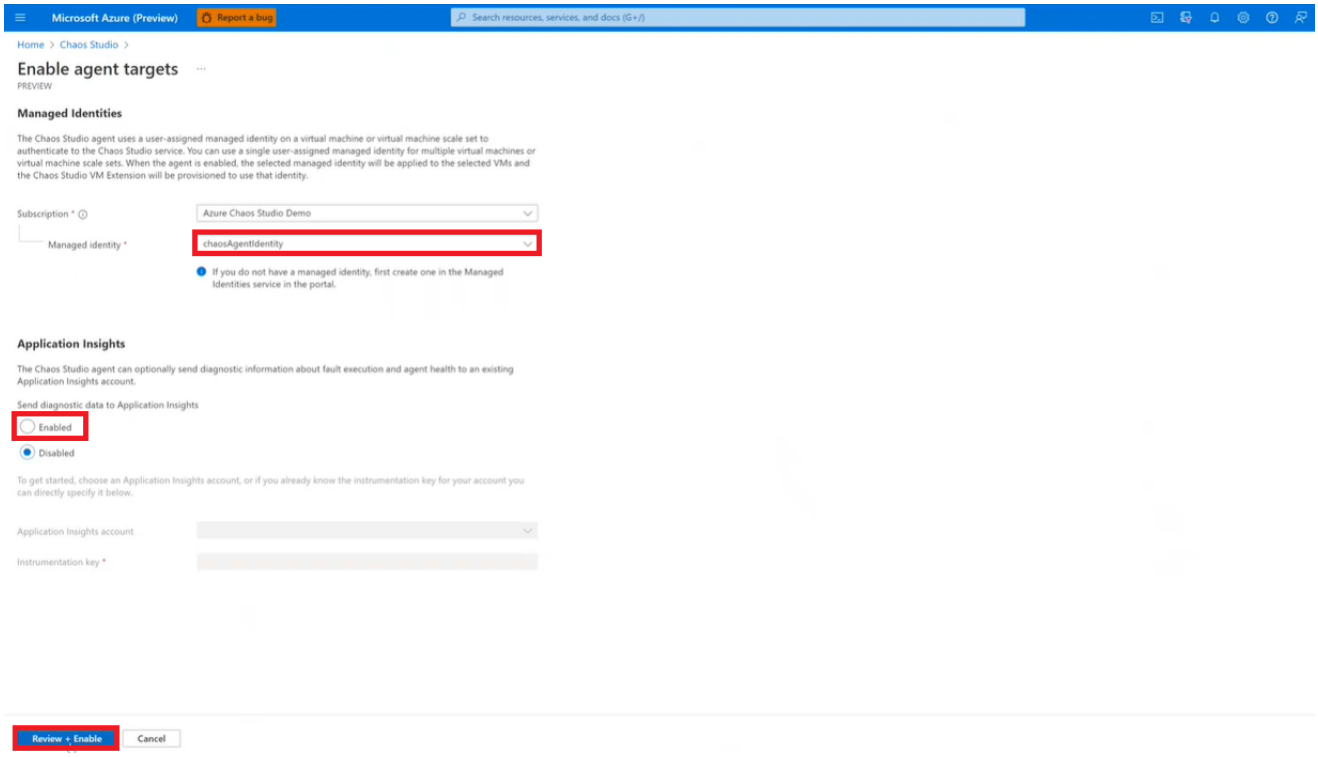
   Note: You will only see resources in regions where Chaos Studio is available (https://azure.microsoft.com/global-infrastructure/services/?products=chaos-studio).



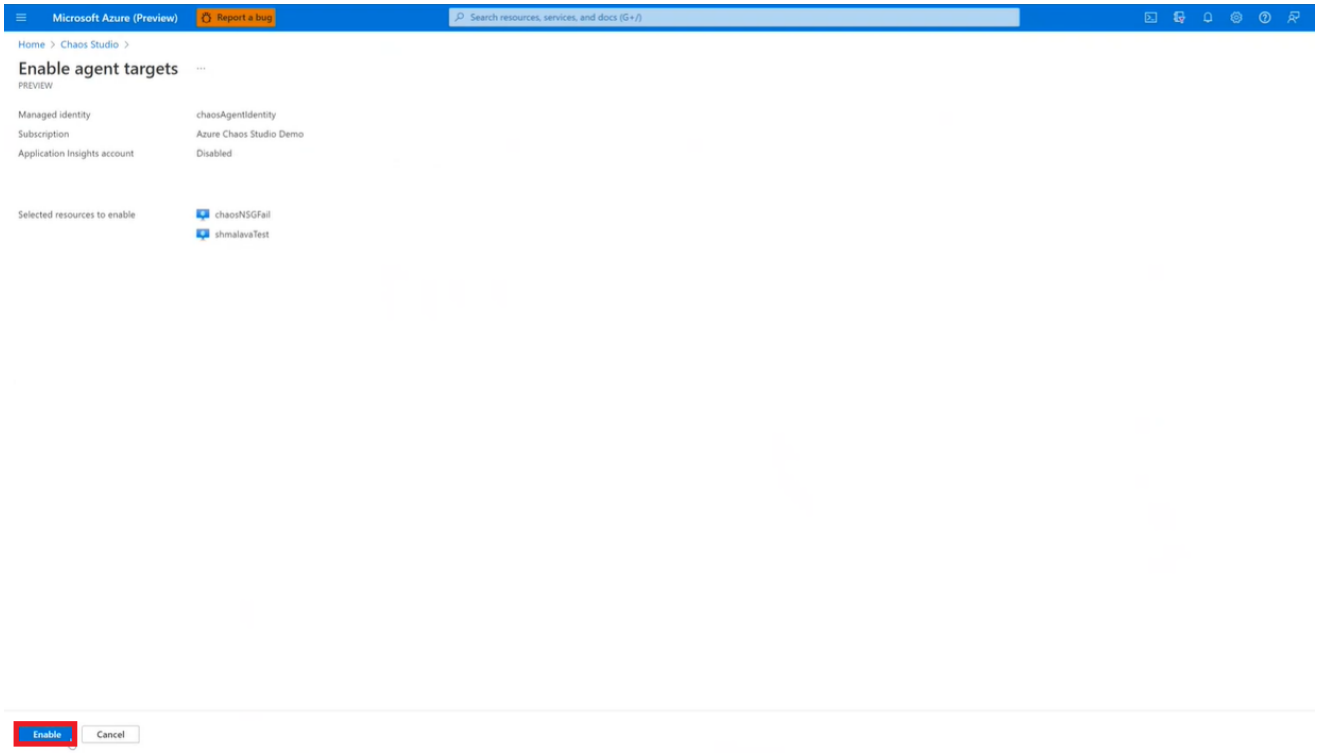
5. Select **Enable targets**, then select **Enable agent-based targets (VM, VMSS)** from the dropdown menu.



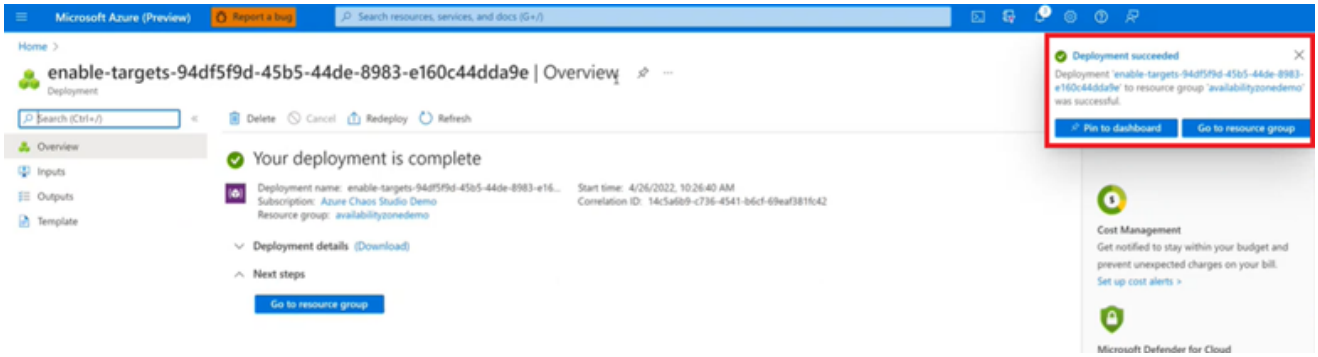
6. You will see the **Enable agent targets** screen. Select your subscription's managed identity from the **Managed identity** dropdown. Optionally, select **Enabled** to send diagnostic data to Application Insights. Select **Review + Enable** when you are ready to proceed.



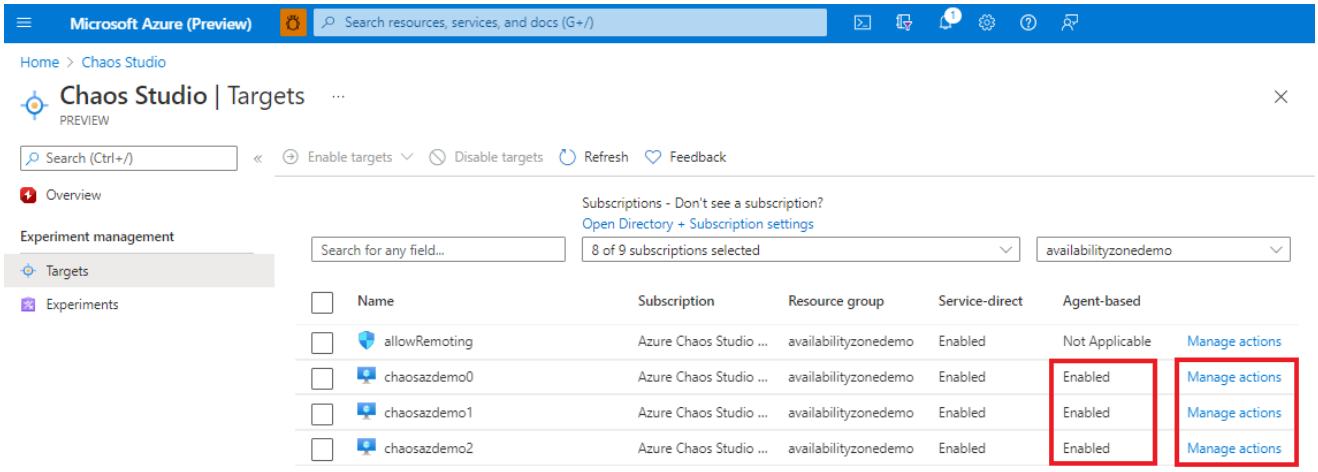
7. Select **Enable**.



8. After a few minutes, you will see a **Deployment succeeded** notification indicating the targets were enabled successfully.

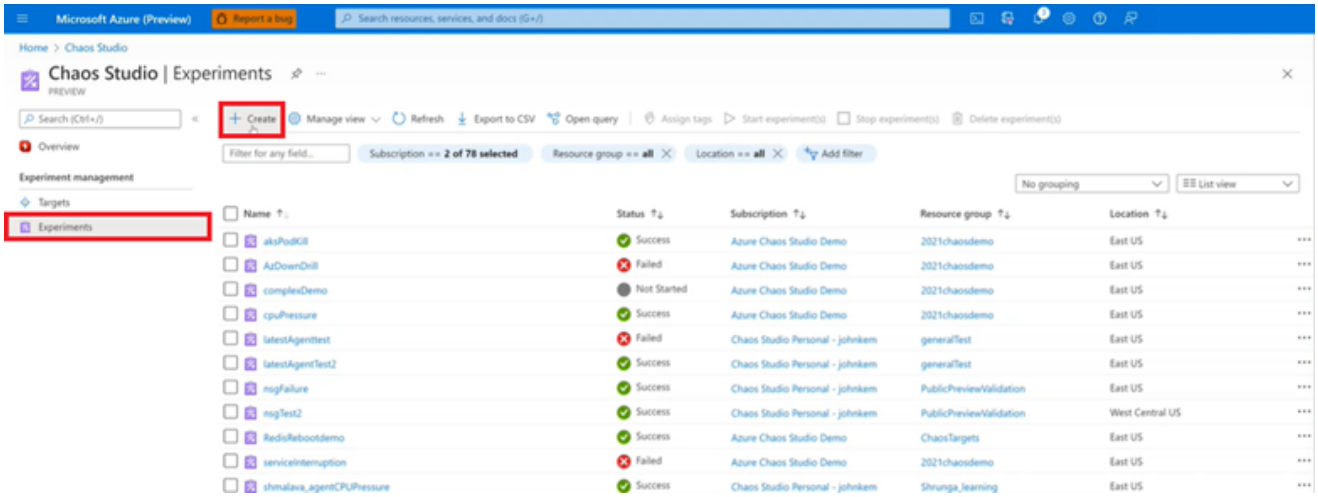


In the **Targets** window, the virtual machine(s) now display **Enabled** under the **Agent-based** heading and display active **Manage actions** links on the right.

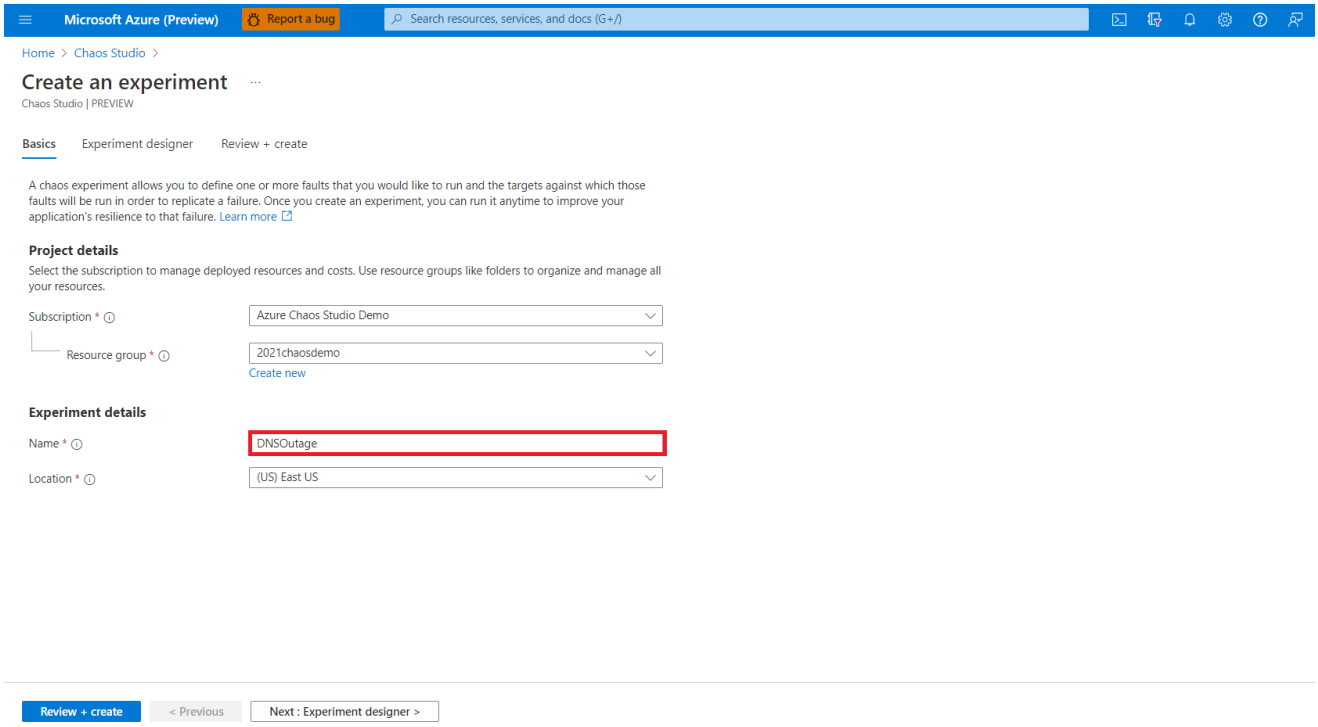


Create the experiment

1. Return to Chaos Studio. Select **Experiments**, then select **Create**.

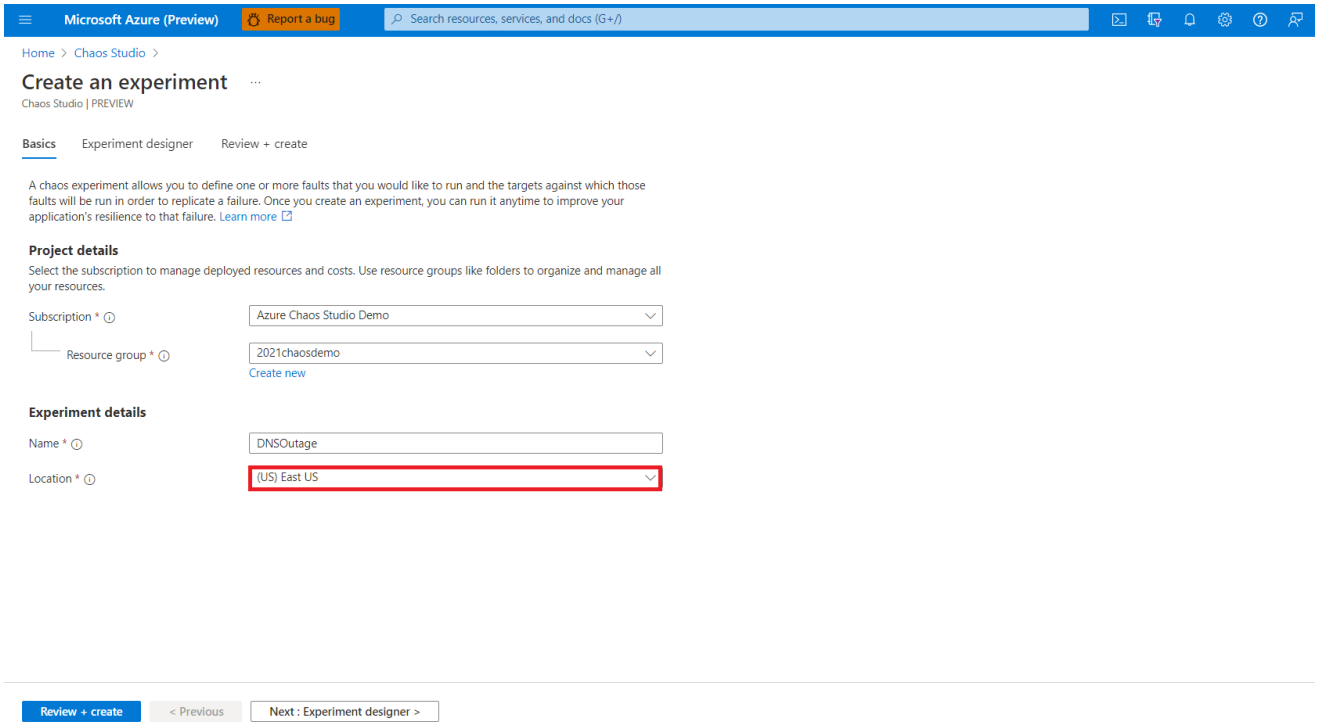


2. You will see the **Create an experiment** screen. In the **Name** field, enter a descriptive name for your experiment.

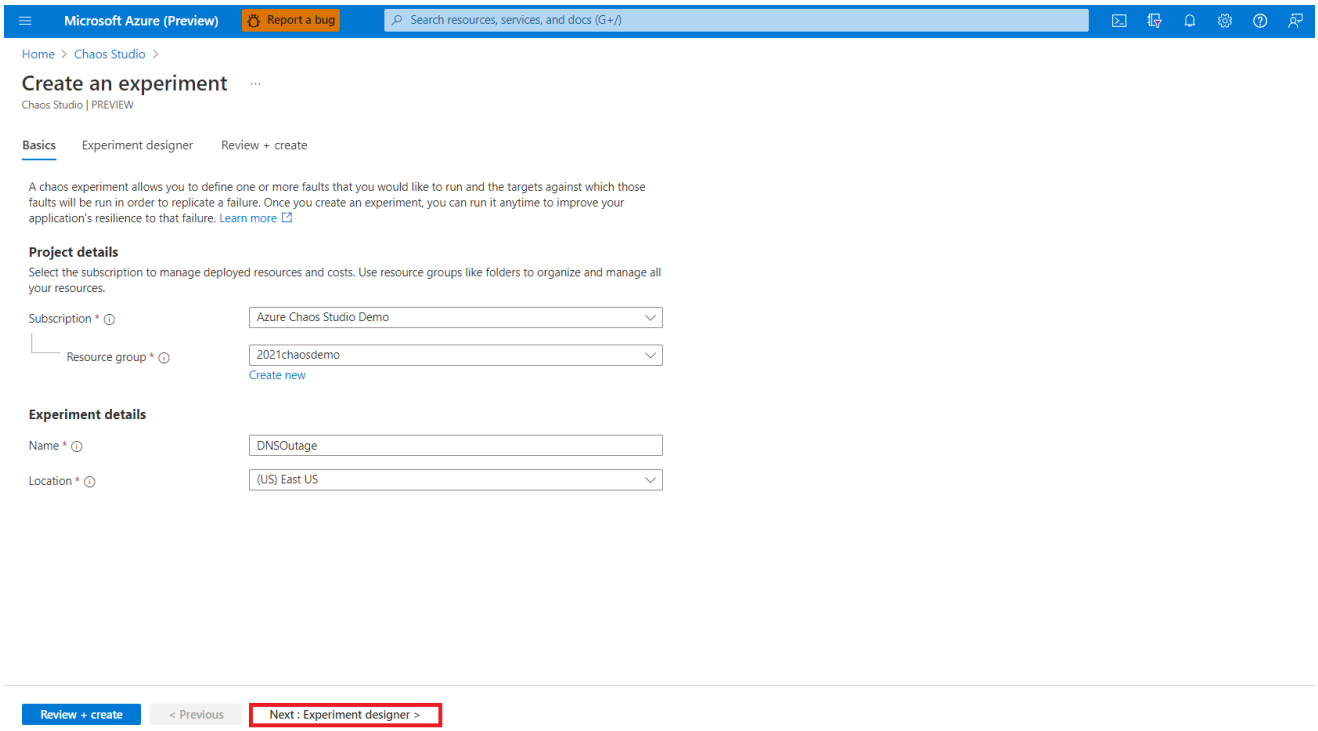


3. Select a region from the **Location** dropdown.

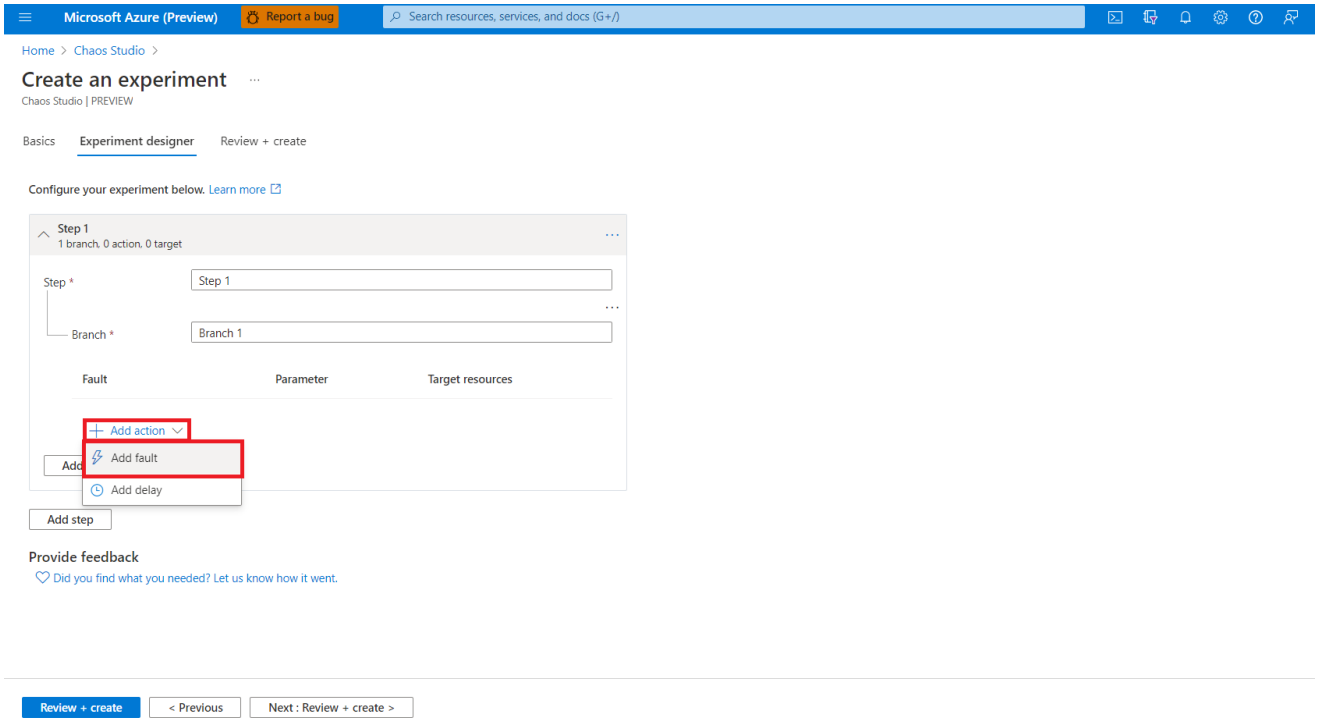
   Note: Chaos Studio must be available in the region you select. Refer to the Products available by region documentation (https://azure.microsoft.com/global-infrastructure/services/?products=chaos-studio) for a list of applicable regions.



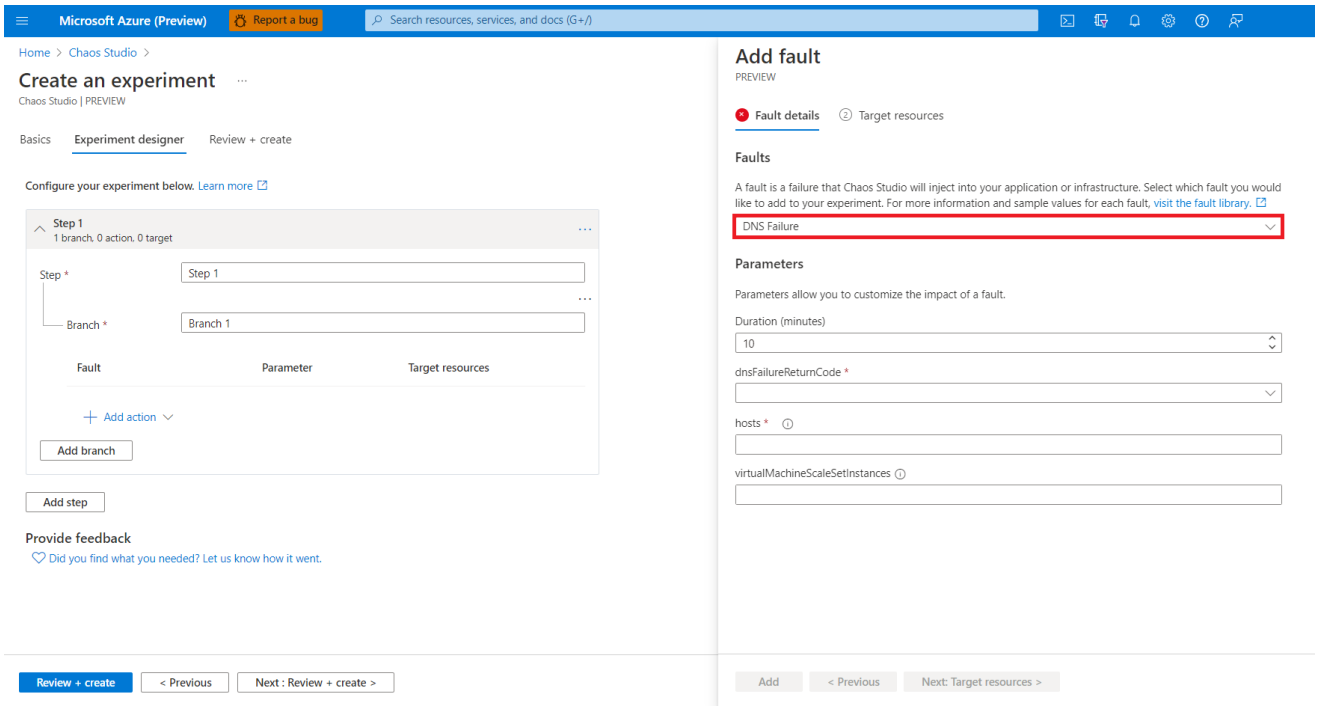
4. Select **Next: Experiment designer**.



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

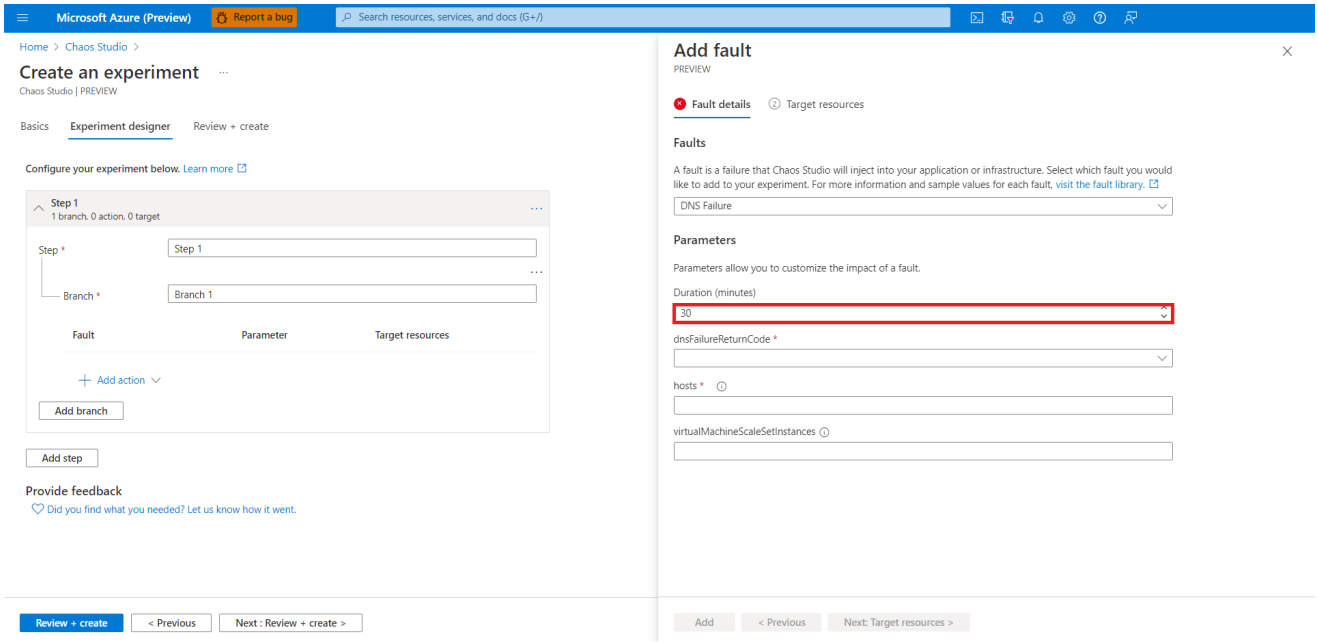


6. The **Add fault** window appears. Select the **Select a fault** dropdown, then select **DNS Failure** from the fault library.



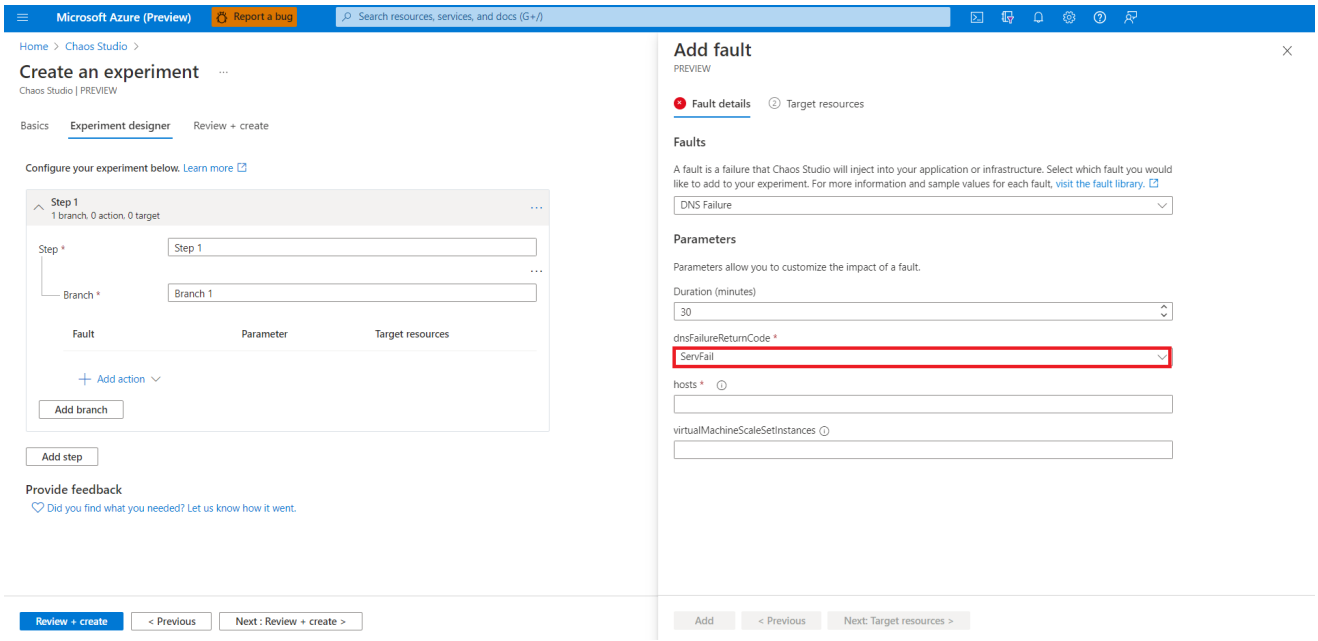
7. You will see a list of parameters specific to the DNS Failure fault. Enter a value for the **Duration (minutes)** parameter to set your desired experiment runtime.

   Note: The duration is set to 10 minutes by default. Although not required, you can change this value to increase or decrease the experiment's runtime. A runtime of 30 to 60 minutes is recommended to allow you to best observe the experiment's impact.

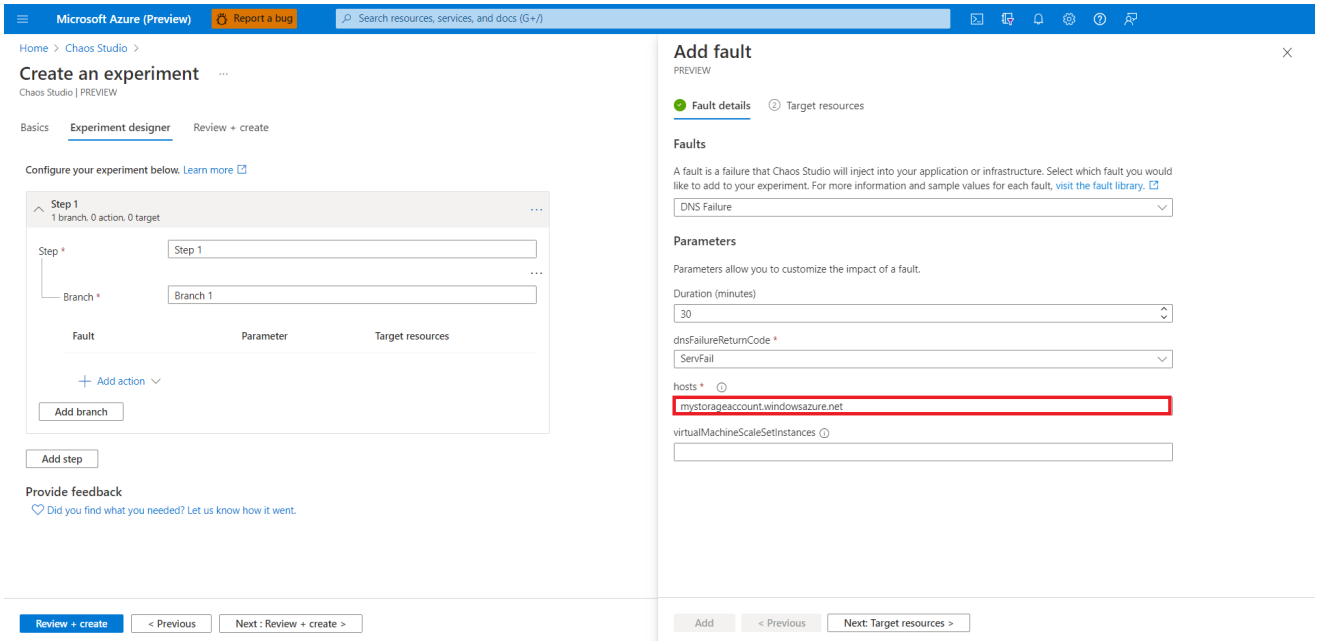


8. Select **ServFail** for the **dnsFailureReturnCode** parameter.

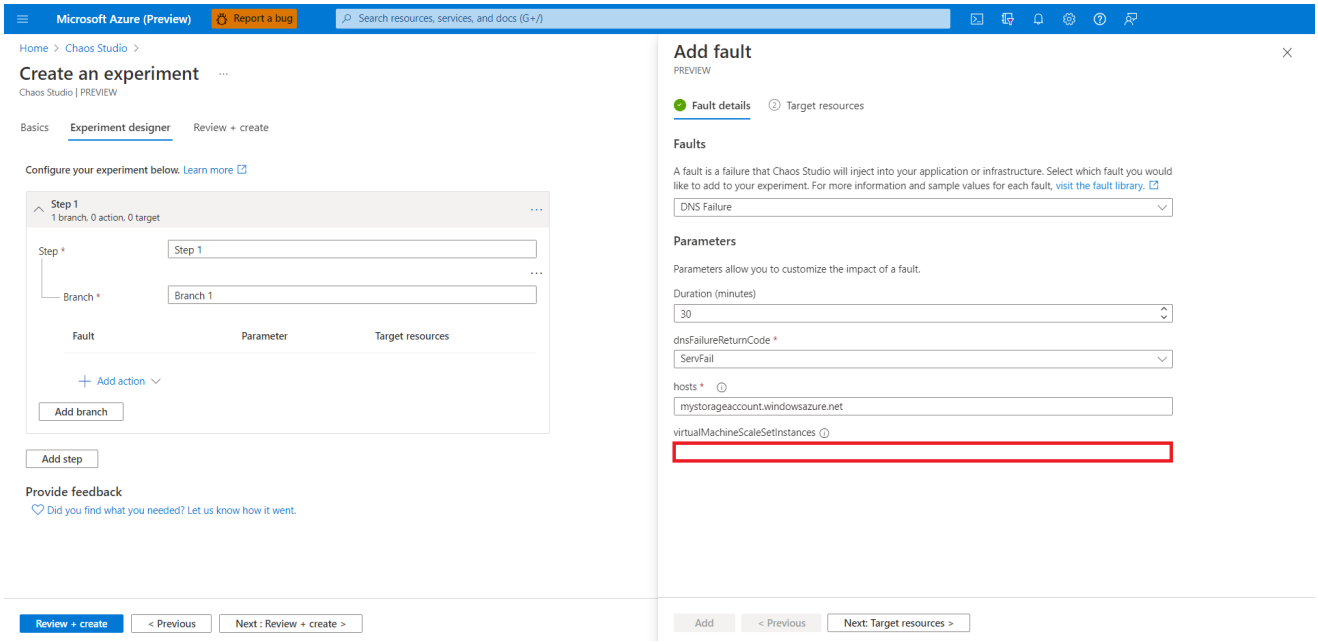
   Note: ServFail indicates the DNS server has an outage, but you may want to use a different parameter depending on the goal of your experiment (https://www.iana.org/assignments/dns-parameters/dns-parameters.xml#dns-parameters-6).



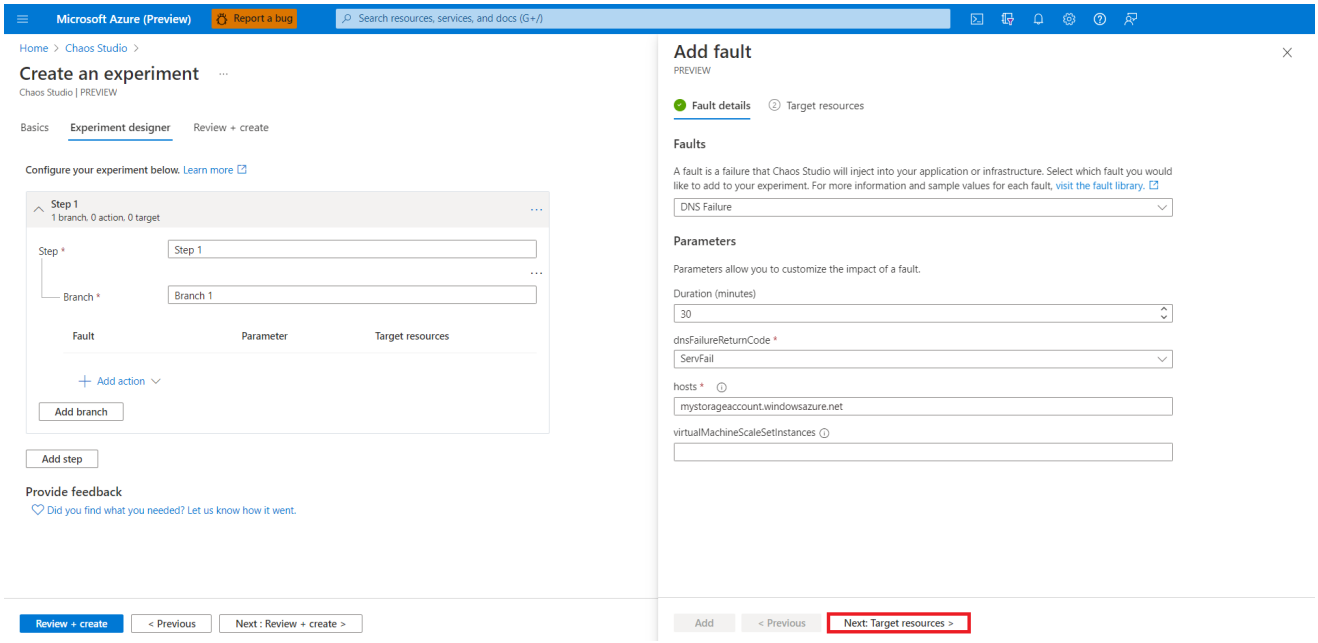
9. For the **hosts** parameter, enter the URL(s) of the resource you are targeting with the DNS outage simulation. If you are targeting multiple URLs, separate them with commas.



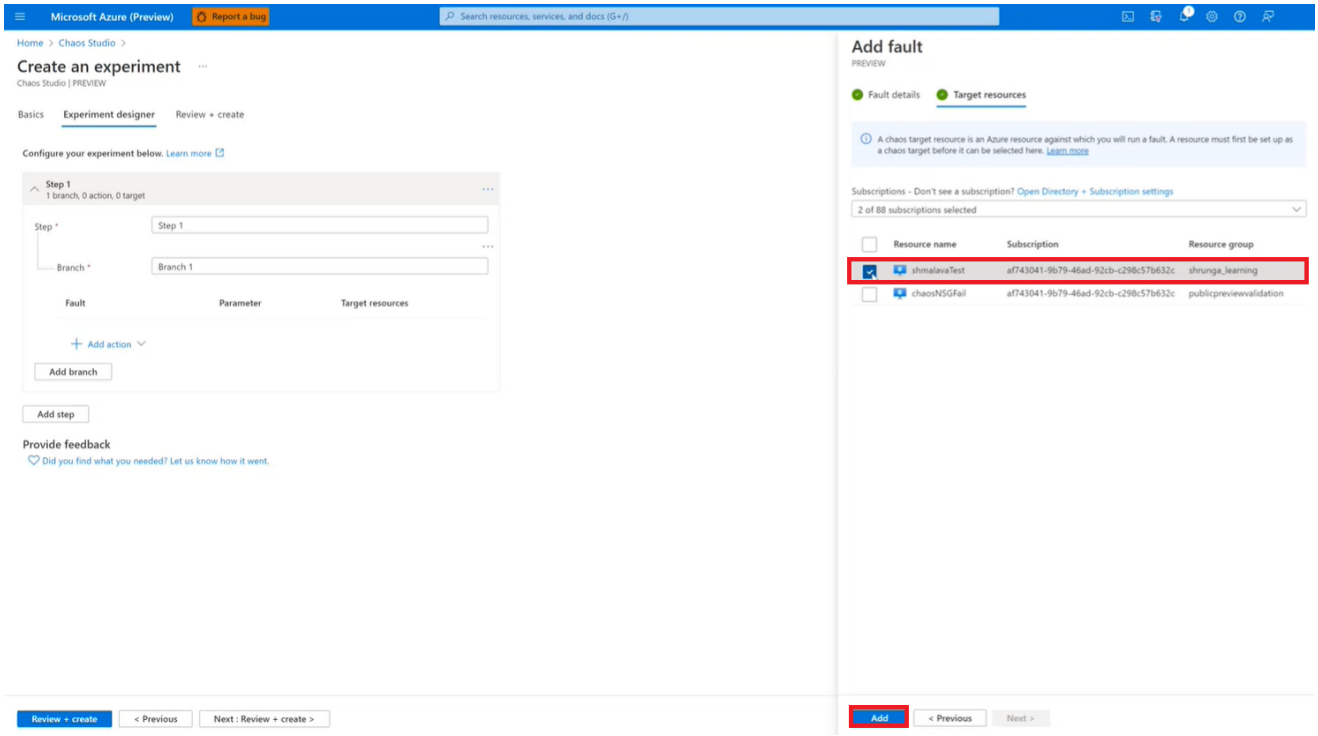
10. If you are using a virtual machine scale set, enter your desired scale set instance numbers, separated by commas, in the **virtualMachineScaleSetInstances** parameter. If you are not using a scale set, leave the parameter blank and continue to the next step.



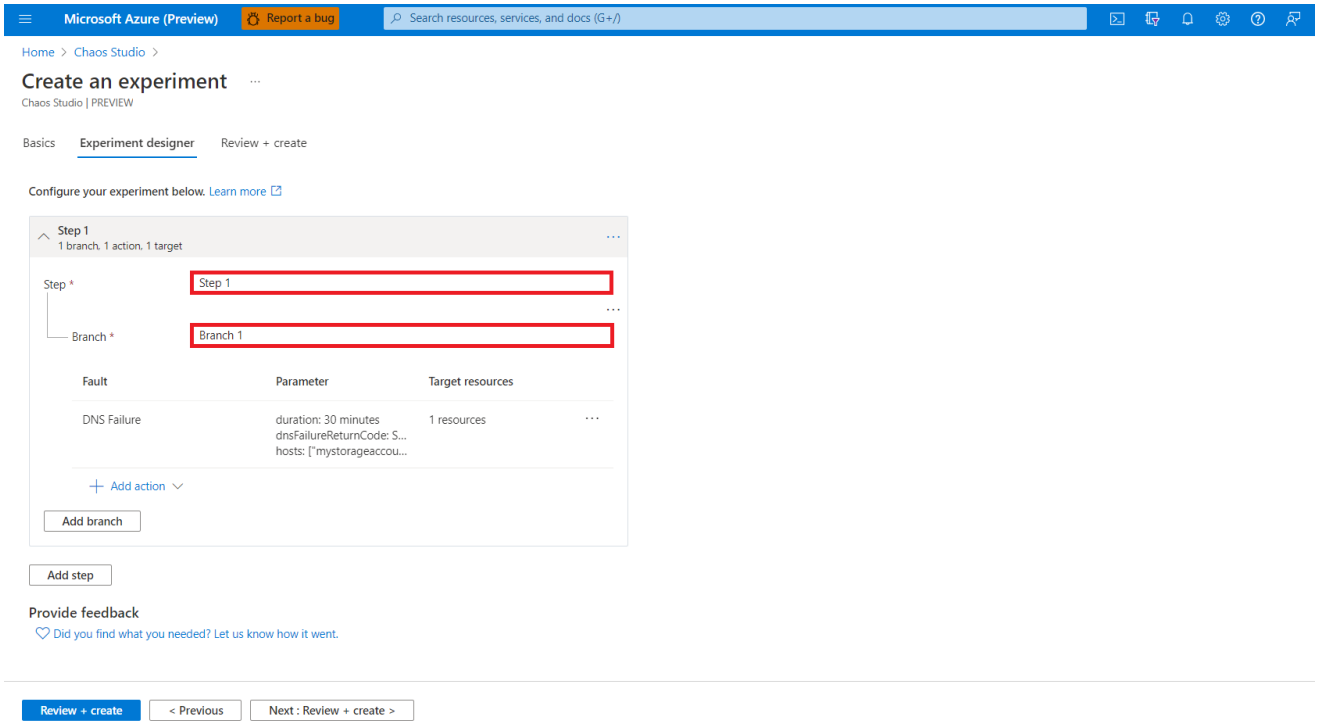
11. Select **Next: Target resources**.



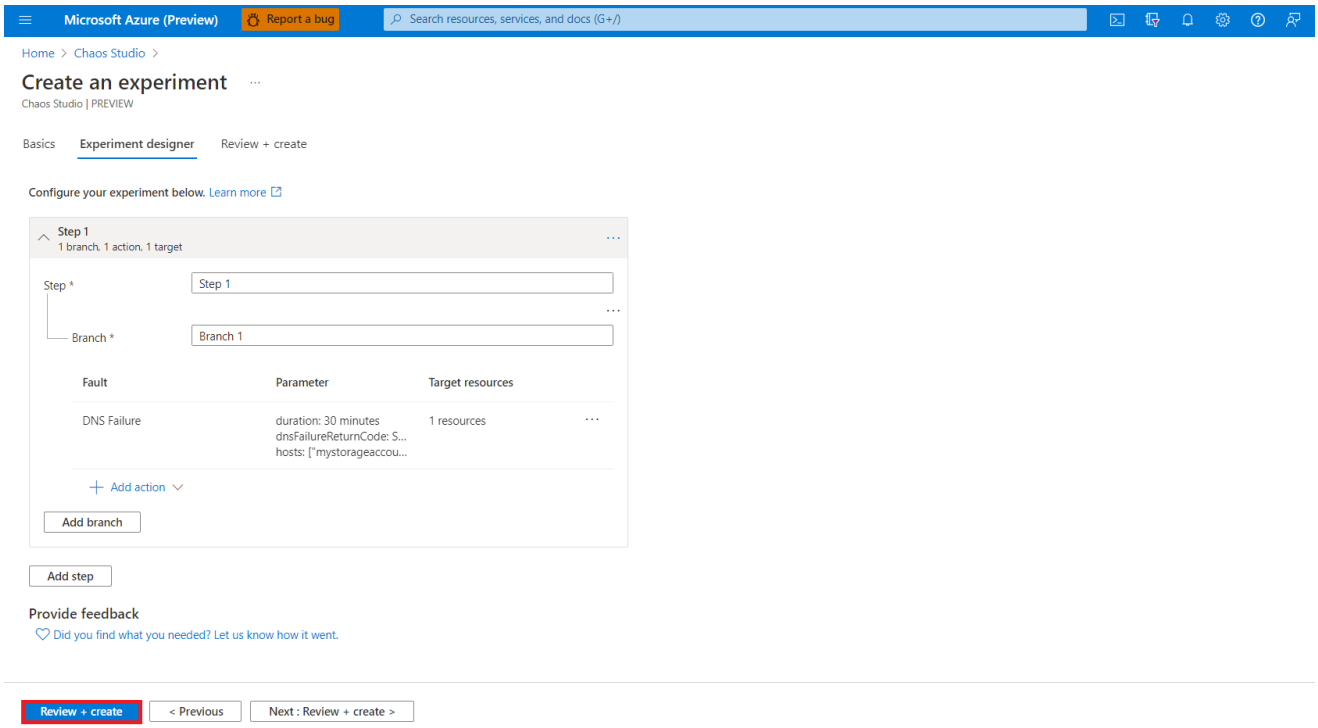
12. Select the virtual machine(s) to apply the experiment to, then select **Add**.



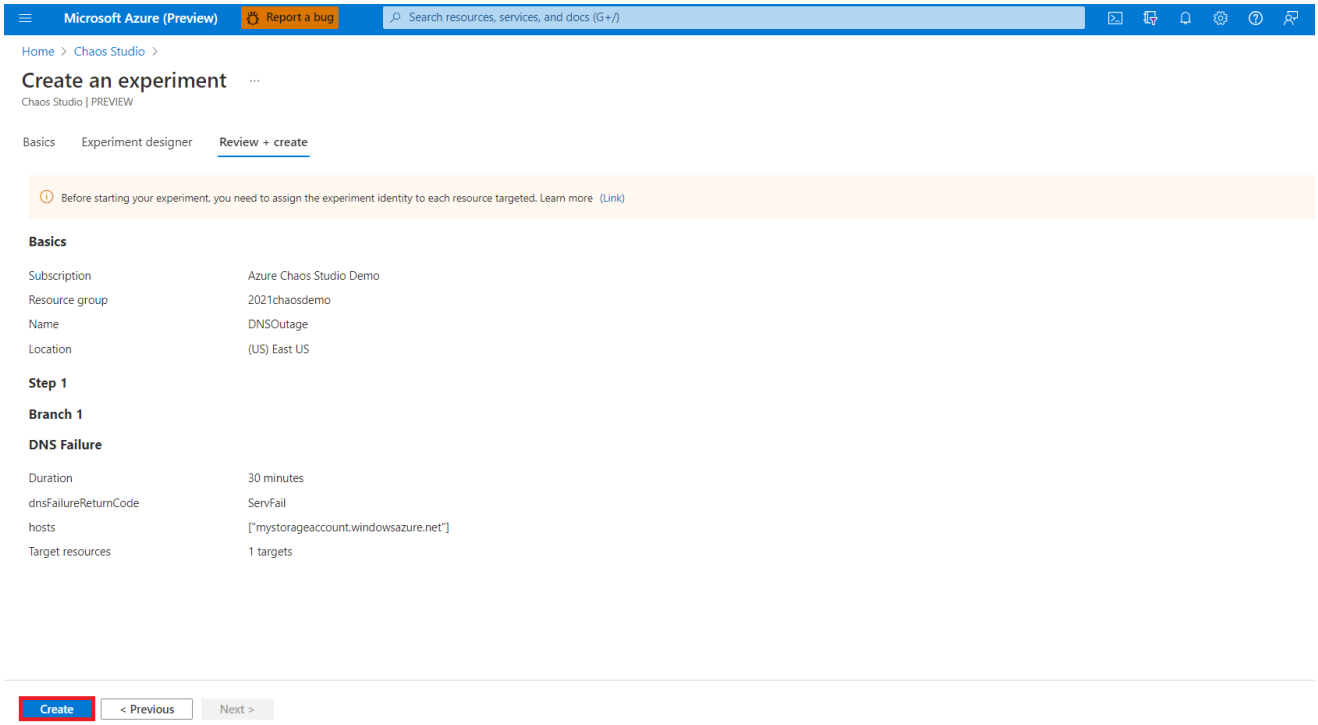
13. Enter descriptive names in the Step field and the Branch field.



14. Select **Review + create**.



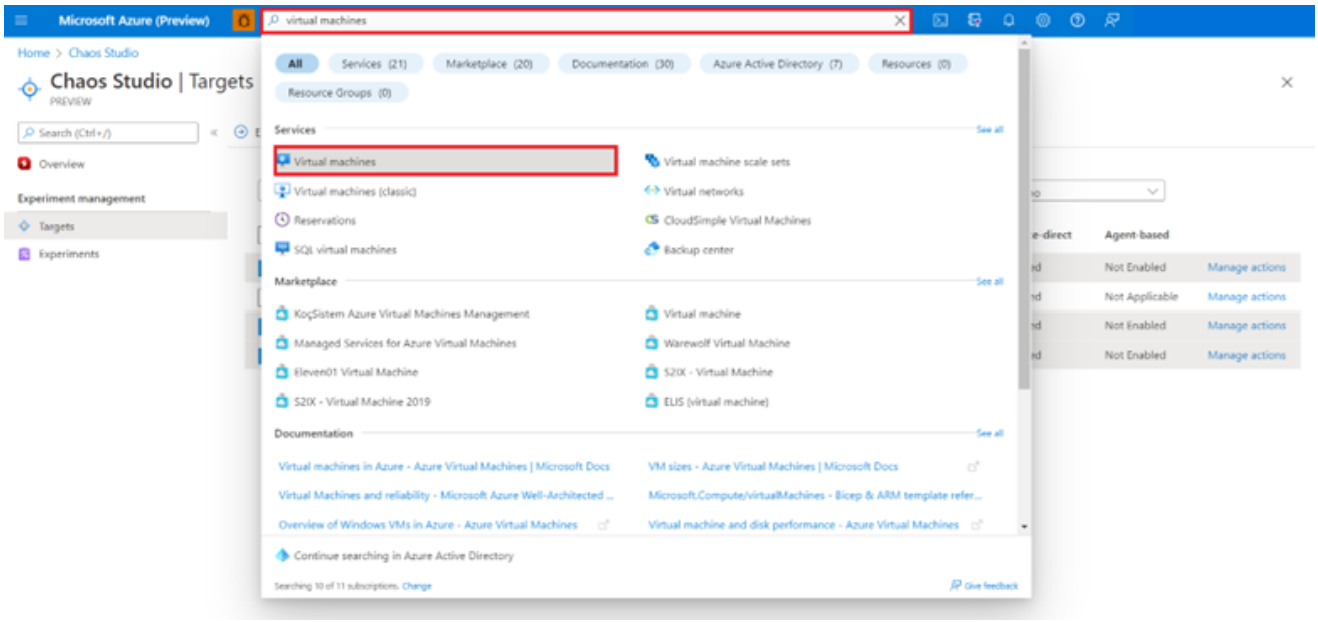
15. The **Review + create** screen appears. Review the experiment details. Then, when you are ready to proceed, select **Create**.



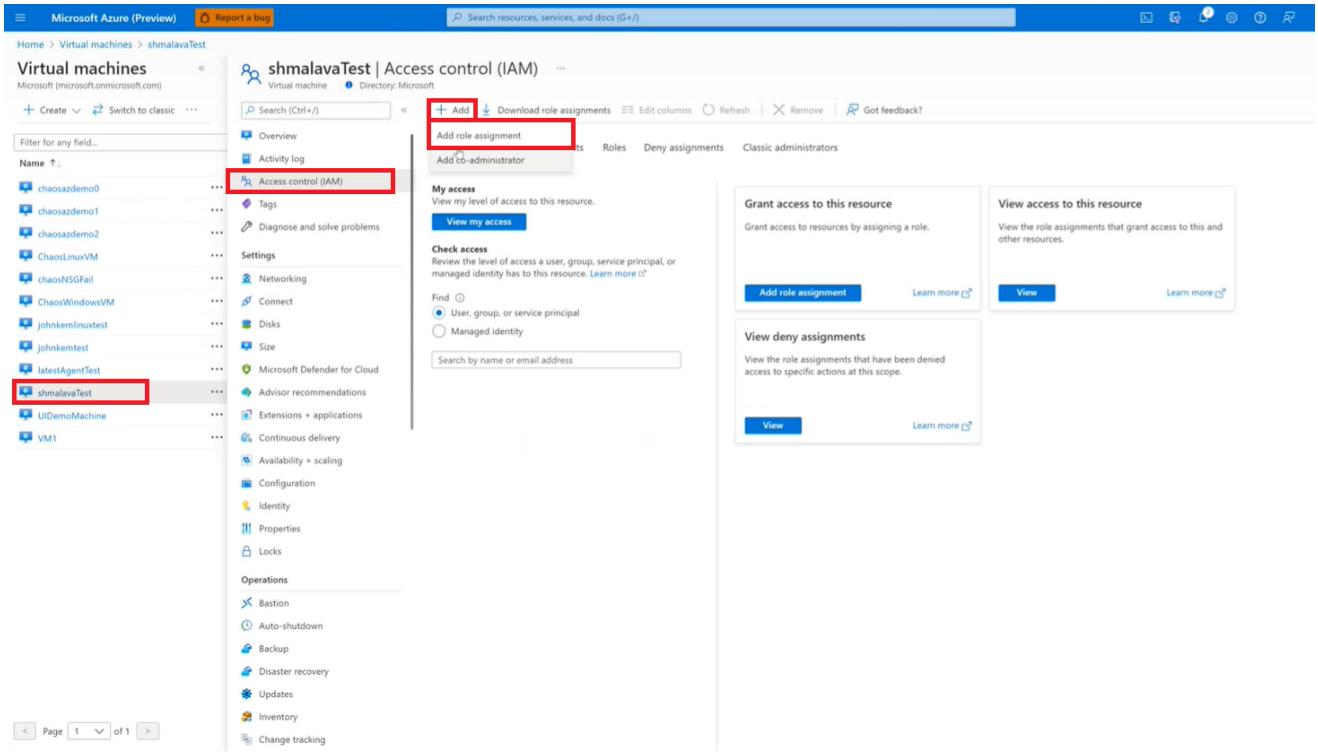
Assign an experiment identity to each targeted resource

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.

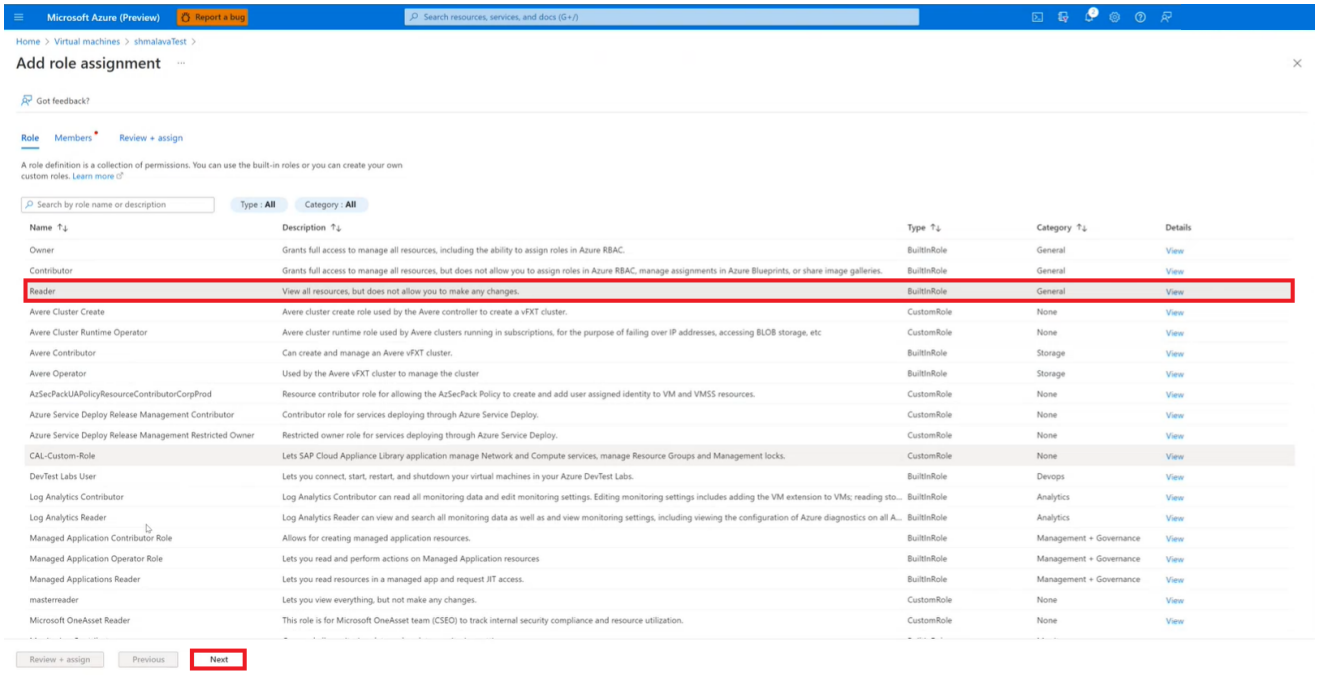
1. Search for **virtual machines** in the search bar and select **Virtual machines** from the menu.



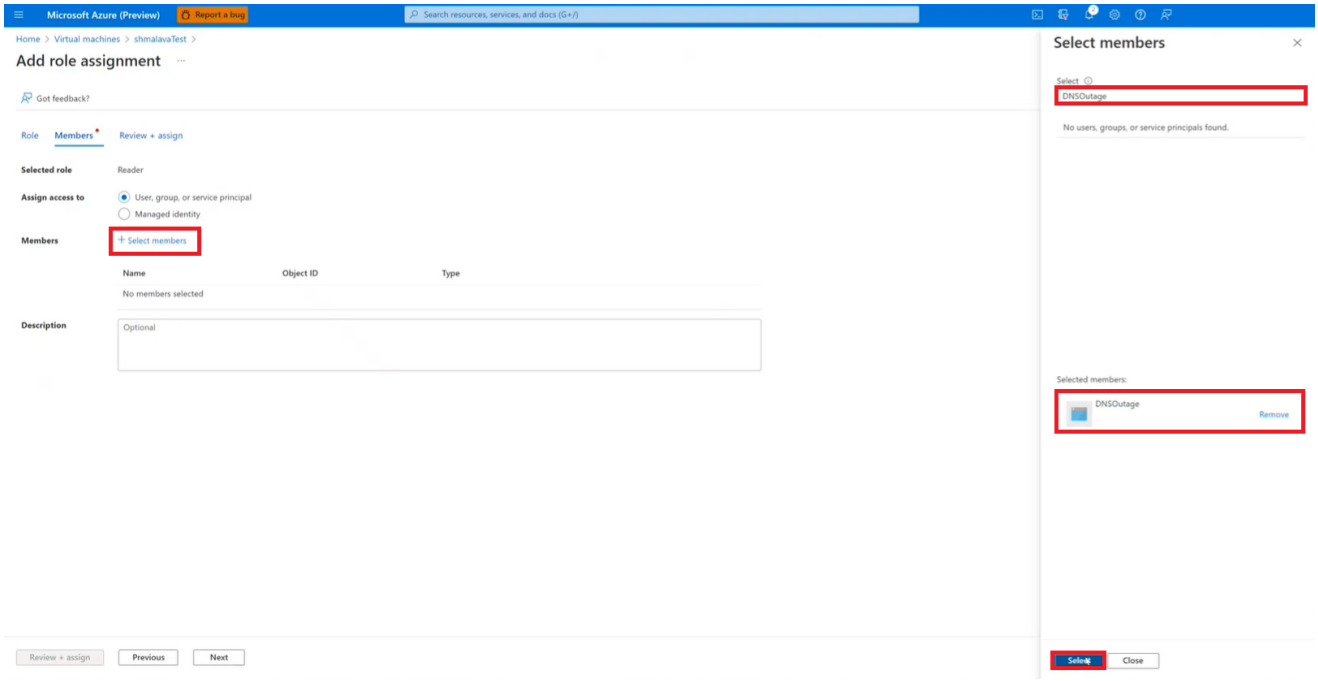
2. Select one of the virtual machines targeted by the created experiment to open its details panel. In this panel, select **Access control (IAM)**, select **Add**, then select **Add role assignment**.



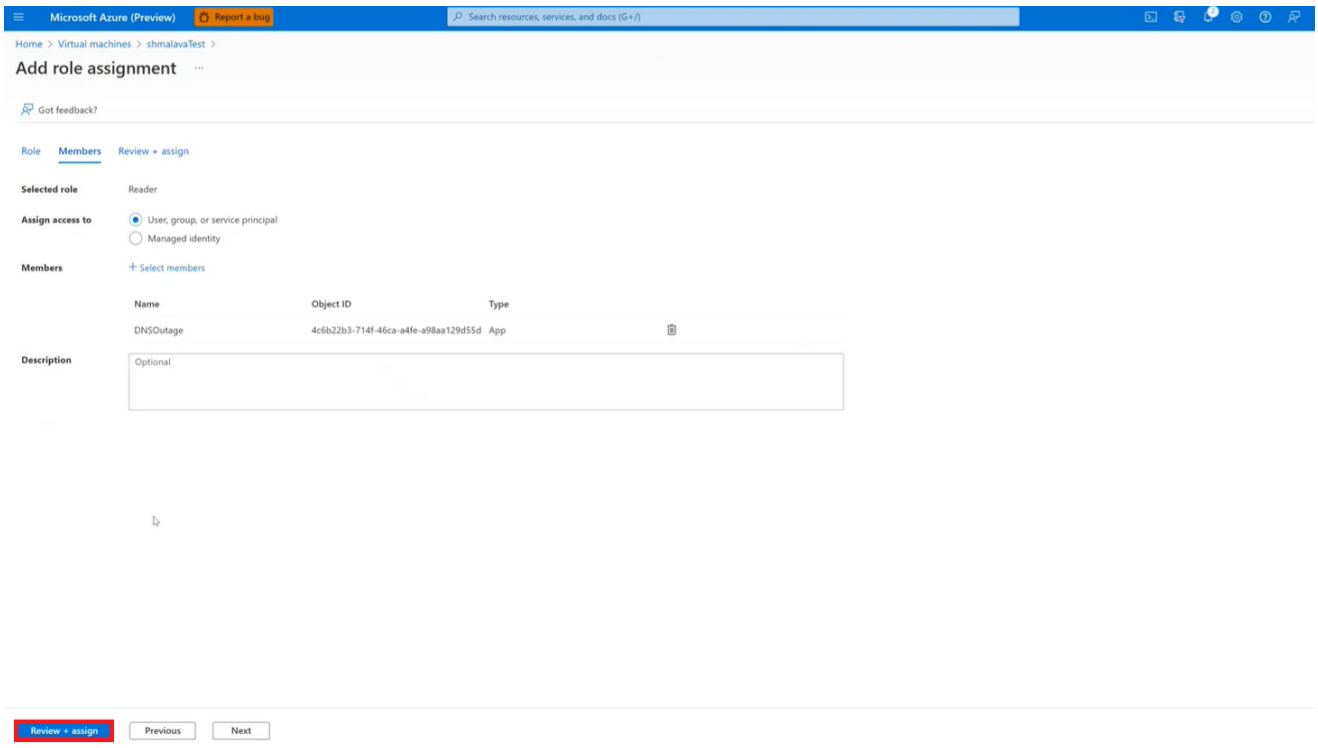
3. Select **Reader** from the list, highlighting it. Select **Next**.

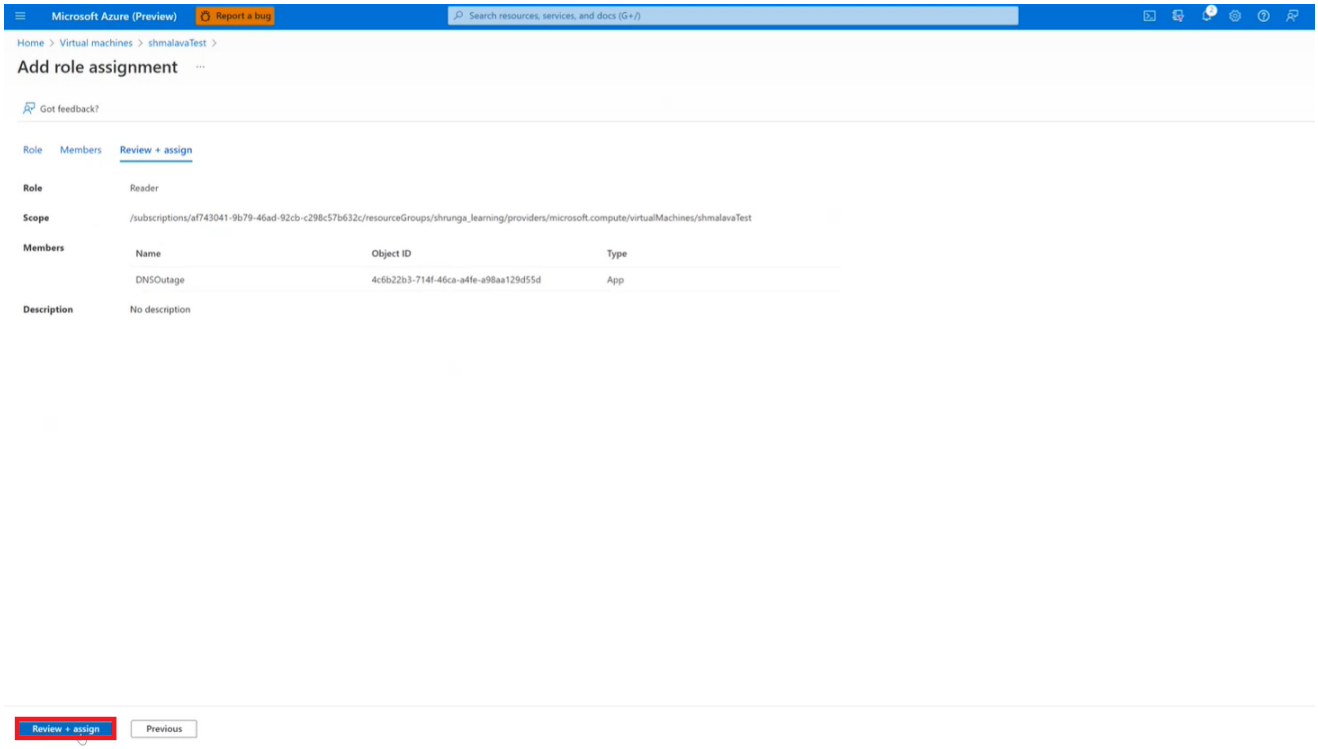


4. Select the **Select members** link. In the **Select** search field in the right sidebar, enter the name of the experiment you created. Select the experiment from the list. Once selected, the experiment will move to the **Selected members:** section. Select the **Select** button.



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

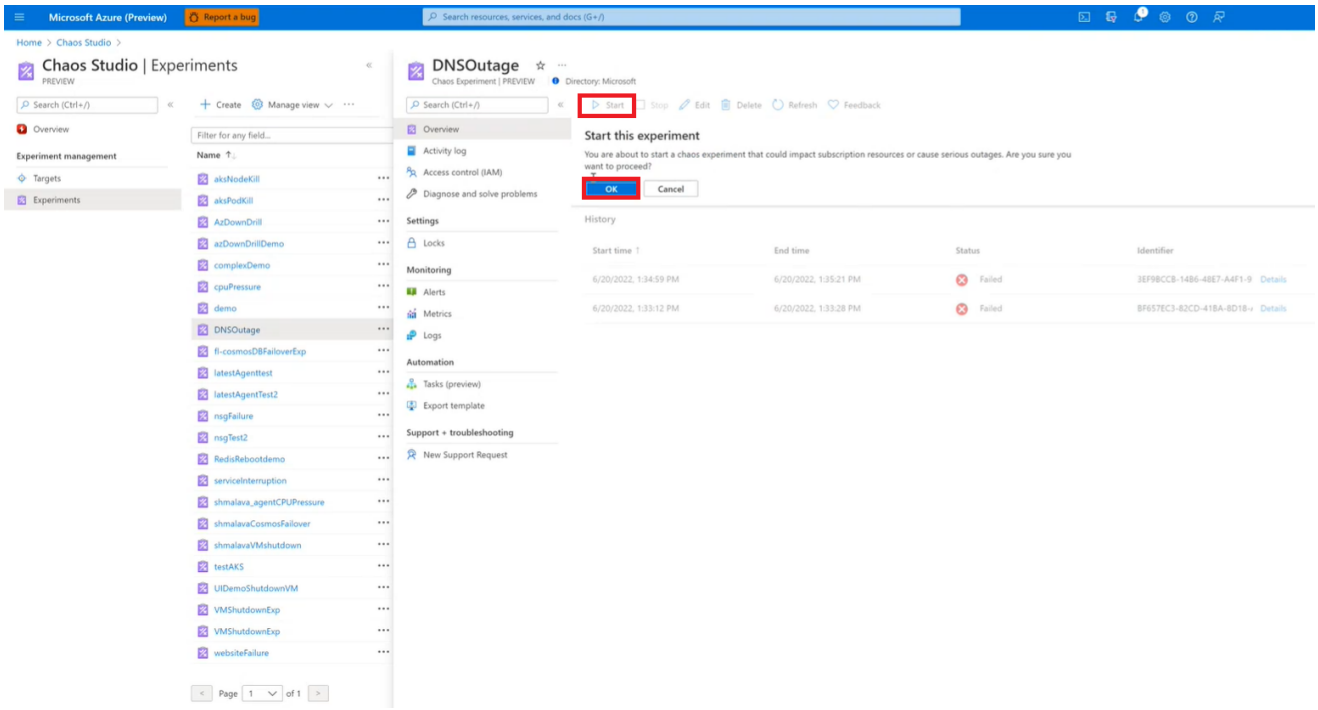




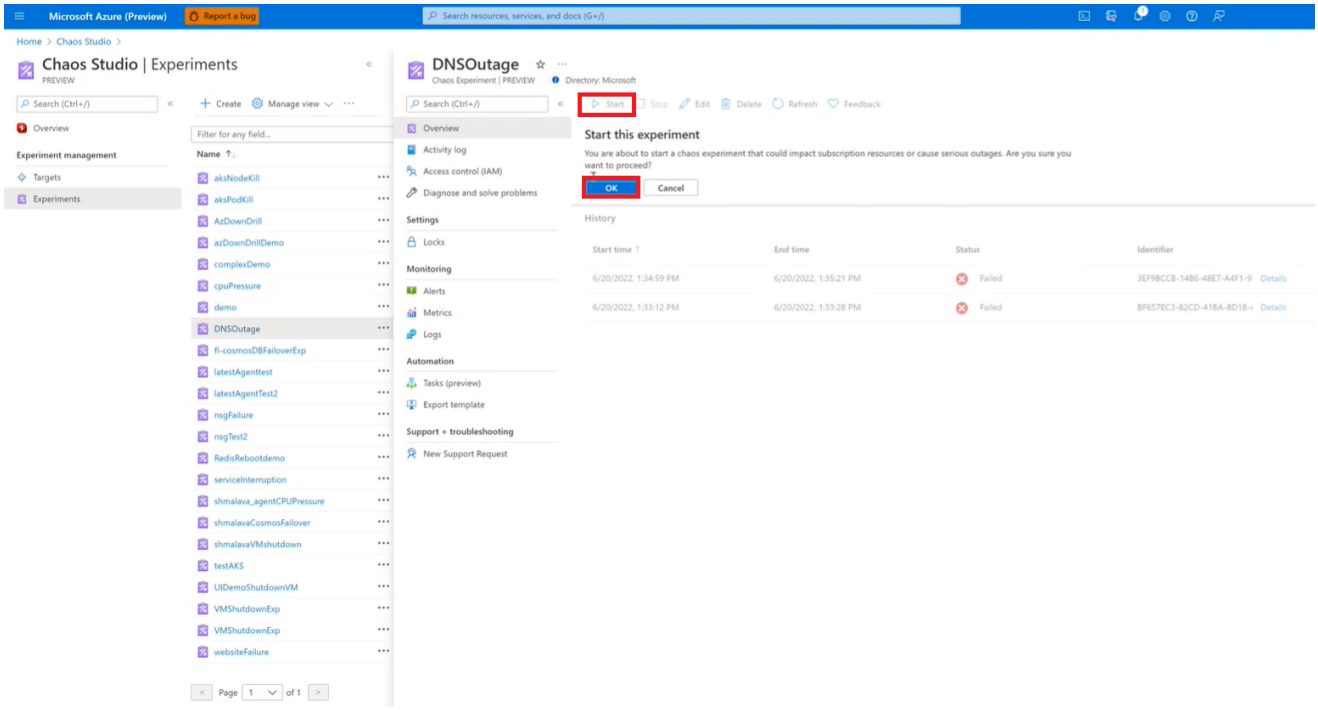
6. Repeat steps 1-6 of this section for each virtual machine or virtual machine scale set targeted in the experiment. If the experiment only targeted one virtual machine, continue to the next section.

Run the experiment

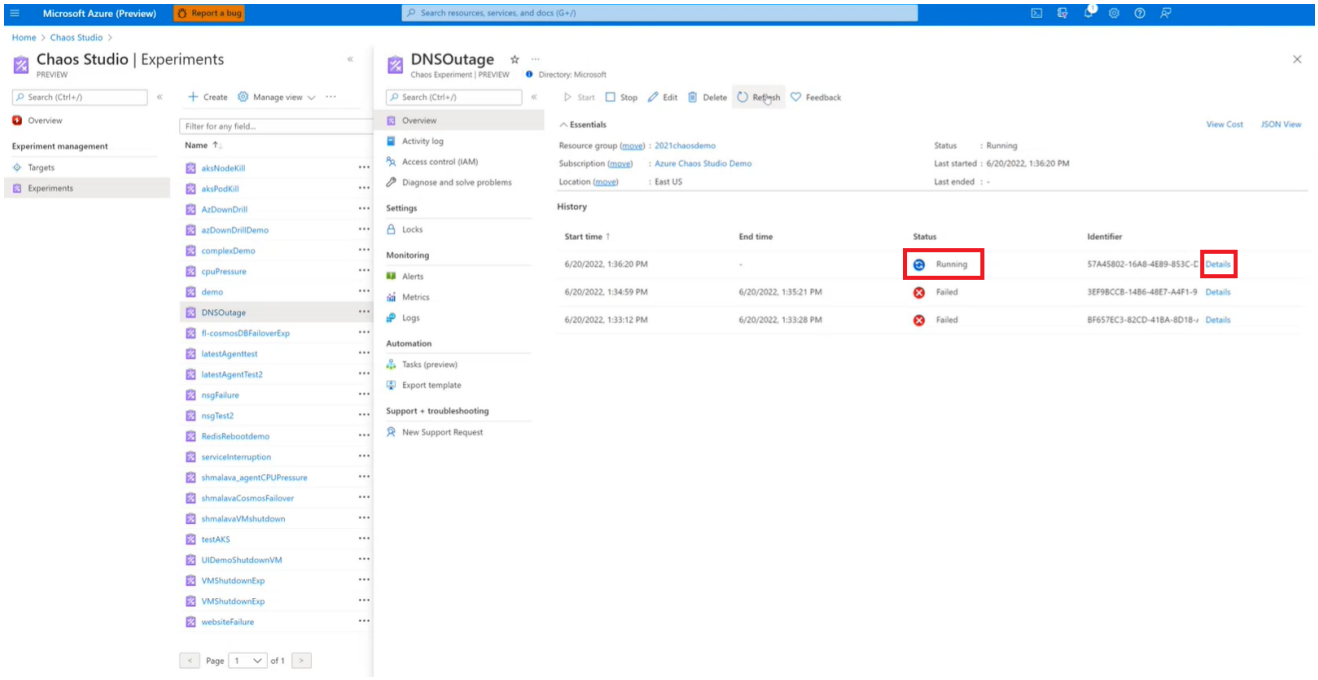
1. From the main Chaos Studio page, select **Experiments**, then select the name of your experiment. You will see the experiment details panel.



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



3. The experiment is successfully running once its status changes to **Running**. 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 example, if your hypothesis addresses availability metrics, analyze your health model in Geneva for the duration of the Chaos experiment to see if there was any impact on the failure rate. If there was an impact, analyze the returned logs and metrics from the experiment to understand why there was a failure rate impact. 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: “In the event of a DNS outage, **ICM incidents were created, and the appropriate resiliency measures were activated because** resilience improvement has been made. I expect to find the experiment results by analyzing __.”

Overview

You have now learned about DNS and observability metrics, how to formulate and evaluate an experiment hypothesis, and how to create an experiment in Azure Chaos Studio that tests the impact of a DNS outage on an application dependency.

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)