

Lab 16: CDN redirectors

Table of Contents

Lab 16: CDN redirectors	1
Goals	1
Requirements.....	1
1. Setting up a CDN redirector in Microsoft Azure:	1
2. Observing the effects of the CDN redirector	12

Goals

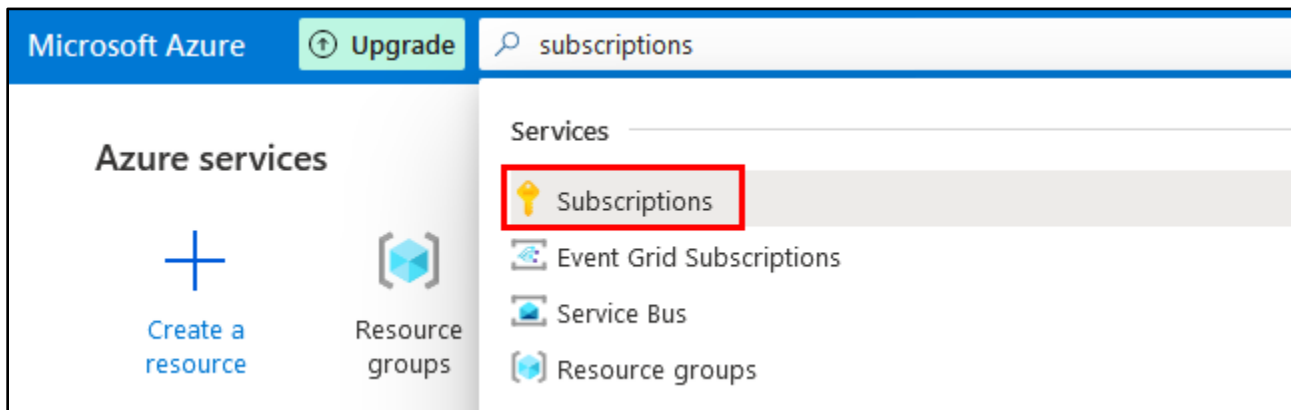
- Use Azure to redirect traffic to C2 infrastructure.

Requirements

- Microsoft Azure trial account.

1. Setting up a CDN redirector in Microsoft Azure:

1. Log in to the Azure portal with your admin account. In the Azure portal, type "subscriptions" in the search bar, and then click on "Subscriptions" in the list of services that appears.



Subscriptions

- On the subscriptions page, click on the subscription you are using for class. If you are using a trial account, you will probably only have one.

Subscriptions

+ Add

View list of subscriptions for which you have role-based access control (RBAC) permissions to manage Azure resources. To Showing subscriptions in [redacted] directory. Don't see a subscription? [Switch directories](#)

My role ⓘ 8 selected Status ⓘ 3 selected

Apply

Showing 1 of 1 subscriptions Show only subscriptions selected in the [global subscriptions filter](#) ⓘ

Search

Subscription name ↑↓	Subscription ID ↑↓	My role ↑↓
Azure subscription 1	[redacted]	Owner

Clicking on the Subscription

- On the subscription page, click on "Resource providers" in the sidebar on the left.

Azure subscription 1

Subscription

Search (Ctrl+/)

Cancel subscription Rename Change directory Feedback

⚠ Your remaining \$171.52 of free credit expires in 23 days. Upgrade to keep going with your account.

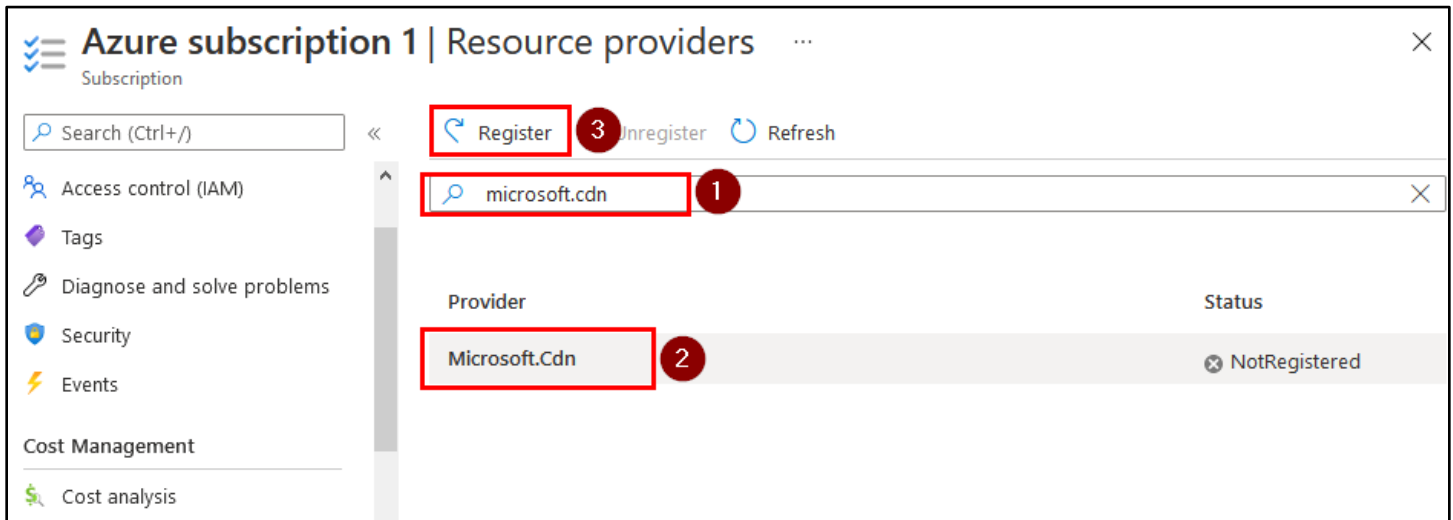
Essentials

Subscription ID	[redacted]	Subscription name	Azure subscription 1
Directory	[redacted]	My role	Owner
Status	Active	Plan	Azure Plan
Parent management group	---	Secure score	Not available

Costs by resource ⓘ [View details >](#) Spending rate and fo [View details >](#)

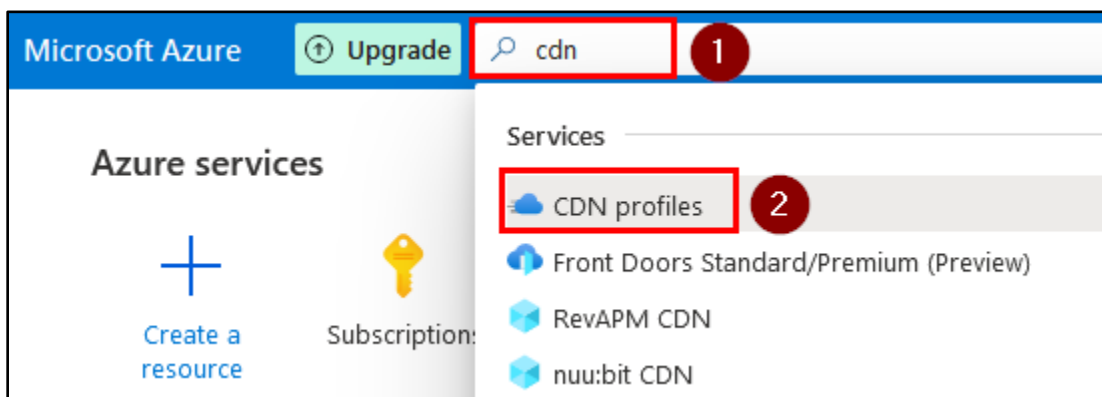
Resource Providers Link

- On the Resource Providers page, type "microsoft.cdn" into the search box. Then click the "Microsoft.cdn" provider that appears. Finally, click the "Register" button near the top of the page to register the provider with your subscription.



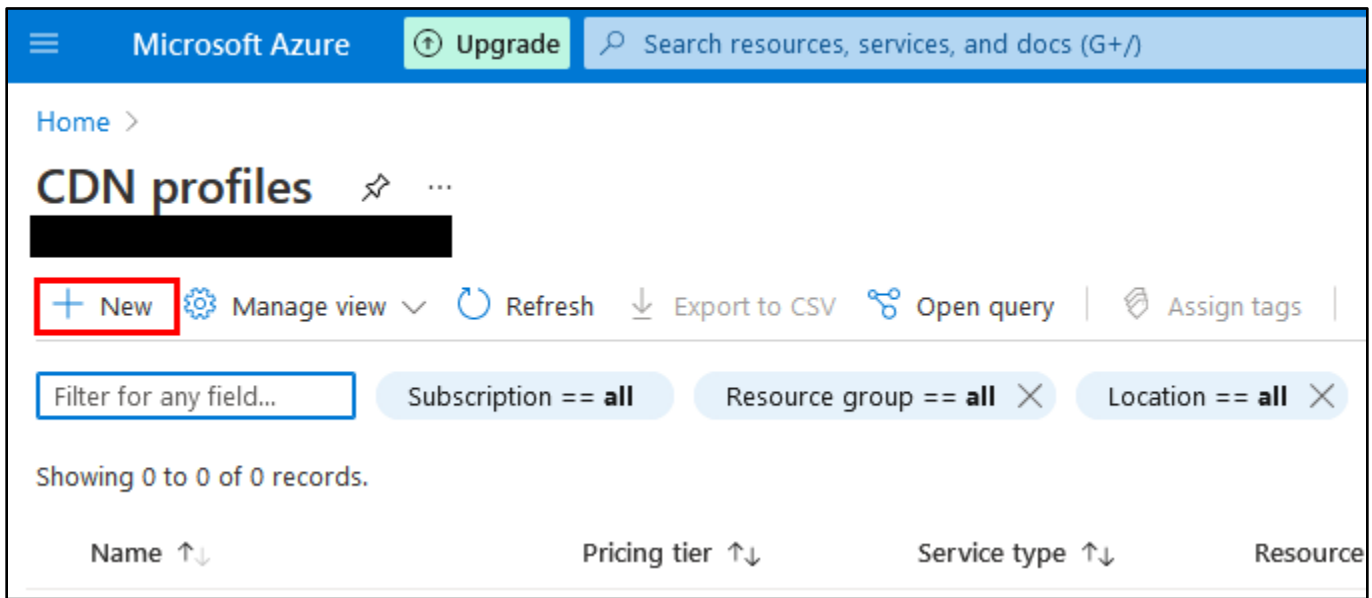
Registering the "Microsoft.cdn" Resource Provider

- Next, type "cdn" into the search box at the top of the Azure website and then click "CDN profiles" in the service results.



CDN Profiles Link

6. On the CDN Profiles page, click the "New" button to create a new CDN profile.



CDN Profile Creation

7. Give your CDN profile a name and assign it to your Azure subscription. Then click "Create new" under the Resource Group box to assign it to a new Resource Group.

A screenshot of the 'Create new' form for a CDN profile. The form has three input fields: 'Name *' with the value 'MyCDNprofile', 'Subscription *' with the value 'Azure subscription 1', and 'Resource group *' which is empty. Below the input fields is a 'Create new' button, which is highlighted with a red box.

New Resource Group Creation

8. Give your new Resource Group a name, and then click OK.

CDN profile ...

Name *
MyCDNprofile ✓

Subscription *
Azure subscription 1 ▾

Resource group *
▾

Create new

A resource group is a container that holds related resources for an Azure solution.

Name *
CDNgroup ✓

OK Cancel

New Resource Group Name

9. Complete the rest of the form by choosing the "Standard Microsoft" pricing tier and then checking the box to create a new CDN endpoint now. When configuring your CDN endpoint, the "CDN endpoint name" will be your redirector's subdomain name on azureedge.net, and it must be unique.

Resource group *
(New) CDNgroup ▾

Create new

Resource group location * ⓘ
East US ▾

Pricing tier (View full pricing details) *
Standard Microsoft ▾

Create a new CDN endpoint now

CDN endpoint name *
teamsapp ✓

.azureedge.net

CDN Profile Configuration

10. For origin type, choose "Custom origin", and then type the target IP address or hostname of your server into the "Origin hostname" box. For this exercise, you can use "lab.adversarydevelopment.com" as your origin hostname. Then click "Create".

CDN profile ... ✕

Name *
 ✓

Subscription *
 ▼

Resource group *
 ▼
[Create new](#)

Resource group location * ⓘ
 ▼

Pricing tier ([View full pricing details](#)) *
 ▼

Create a new CDN endpoint now

CDN endpoint name *
 ✓
.azureedge.net

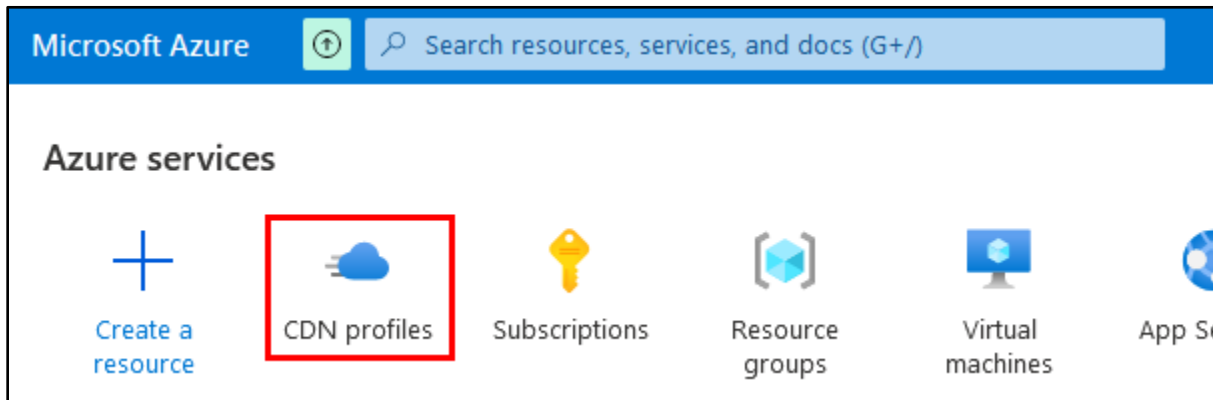
Origin type *
 ▼

Origin hostname * ⓘ
 ✓

[Automation options](#)

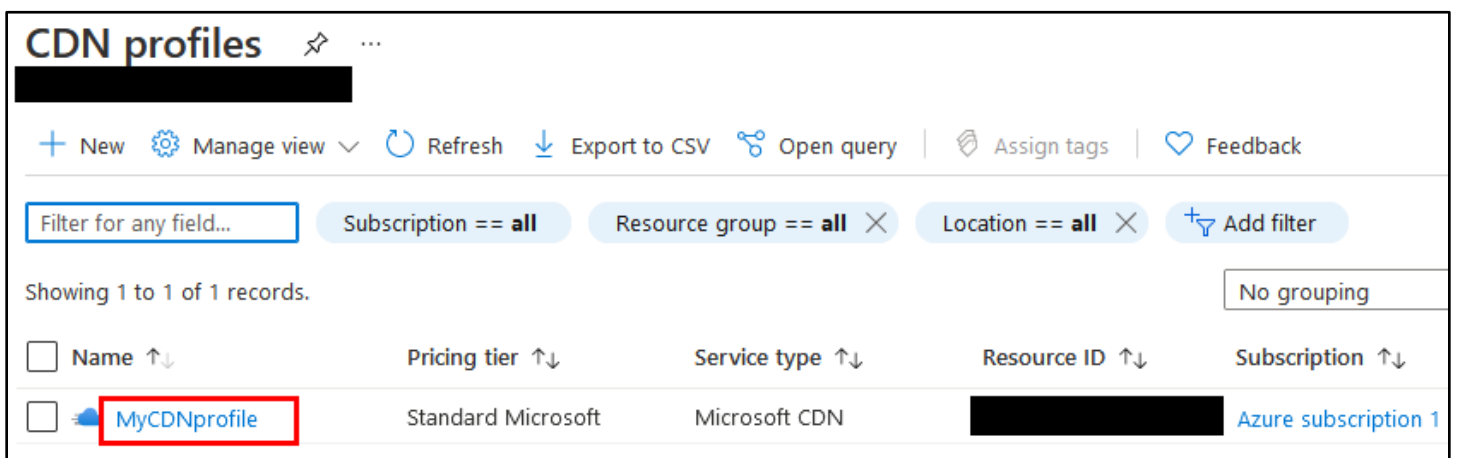
Custom Origin Specification

11. After creating the CDN profile, you'll be redirected back to the Azure home page. Click "CDN profiles" at the top to go back to the CDN profiles page.



CDN Profiles Icon

12. On the CDN profiles page, click the name of the CDN profile you just setup.



Clicking the CDN Profile

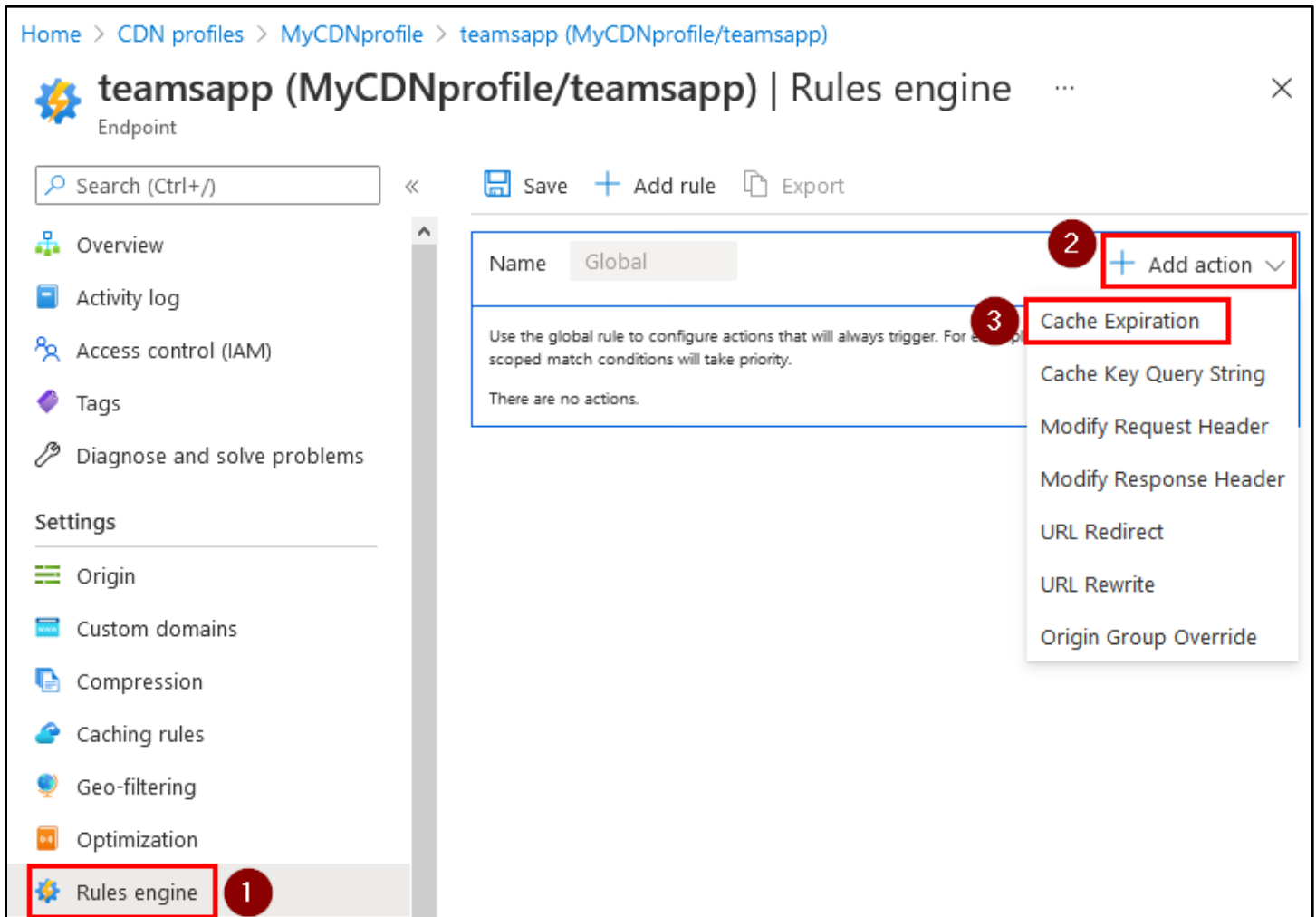
13. On your CDN profile page, click on the endpoint you just created.

The screenshot shows the Azure portal interface for a CDN profile named 'MyCDNprofile'. The left sidebar contains navigation options: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Properties, Quickstart, and Locks. The main content area is divided into 'Essentials' and 'Endpoints' sections. The 'Essentials' section displays metadata such as Resource group (CDNgroup), Status (Active), and Subscription (Azure subscription 1). The 'Endpoints' section contains a table with the following data:

Hostname	Status	Protocol	Origin type
teamsapp.azureedge.net	Running	HTTP, HTTPS	Custom origin

Selecting the Endpoint

14. In the sidebar on the left, click on "Rules engine" to configure global caching rules for your CDN endpoint. Then click "Add action" on the right and "Cache Expiration" in the drop-down menu.



Rules Engine Configuration

15. Under "Cache behavior", choose "Bypass cache", and then click Save at the top.

Home > CDN profiles > MyCDNprofile > teamsapp (MyCDNprofile/teamsapp)

teamsapp (MyCDNprofile/teamsapp) | Rules engine

Endpoint

Search (Ctrl+/) << Save Add rule Export

Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems

Settings
Origin
Custom domains
Compression

Name: Global + Add action

Use the global rule to configure actions that will always trigger. For example, cache TTL settings. Later rules with scoped match conditions will take priority.

Always
Cache expiration
Cache behavior *

Bypass cache 1

Hours Minutes Seconds

Bypass Cache

16. After a few moments, you should see a message stating that your endpoint's configuration has been updated.

Microsoft Azure Search resources, services, and docs (G+)

Home > CDN profiles > MyCDNprofile > teamsapp (MyCDNprofile/teamsapp)

teamsapp (MyCDNprofile/teamsapp) | Rules engine

Endpoint

Search (Ctrl+/) << Save + Add rule Export

Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems

Settings

Name: Global + Add action

Use the global rule to configure actions that will always trigger. For example, cache TTL settings. Later rules with scoped match conditions will take priority.

Always
Cache expiration
Cache behavior *

Bypass cache

Days Hours Minutes Seconds

Successfully updated the endpoint's configuration... 7:25 PM
It can take up to 10 minutes for endpoint 'teamsapp' settings to propagate.

Configuration Updated Successfully

17. Next, click on "Caching rules" in the sidebar on the left. Then click the dropdown to change "Query string caching behavior" to "Bypass caching for query strings". Finally, click the Save button at the top.

Home > CDN profiles > MyCDNprofile > teamsapp (MyCDNprofile/teamsapp)

Endpoint

Search (Ctrl+/)

3 Save Discard

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Origin

Custom domains

Compression

1 Caching rules

About This Feature

Control how CDN caches your content and how unique query strings are handled.

[Learn more](#)

Query string caching behavior ⓘ Bypass caching for query strings ^

Ignore query strings

2 Bypass caching for query strings

Cache every unique URL

Configuring Caching Rules

18. Once again, after a few moments you should receive a message stating that your endpoint's configuration has been updated.

Home > CDN profiles > MyCDNprofile > teamsapp (MyCDNprofile/te...

Endpoint

Search (Ctrl+/)

Save Discard

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

About This Feature

Control how CDN caches your content and how unique query strings are handled.

[Learn more](#)

Query string caching behavior ⓘ Bypass caching for query strings v

Successfully updated the endpoint's confi... 7:27 PM
It can take up to 10 minutes for endpoint 'teamsapp' settings to propagate.

Successful Change to Caching Rules

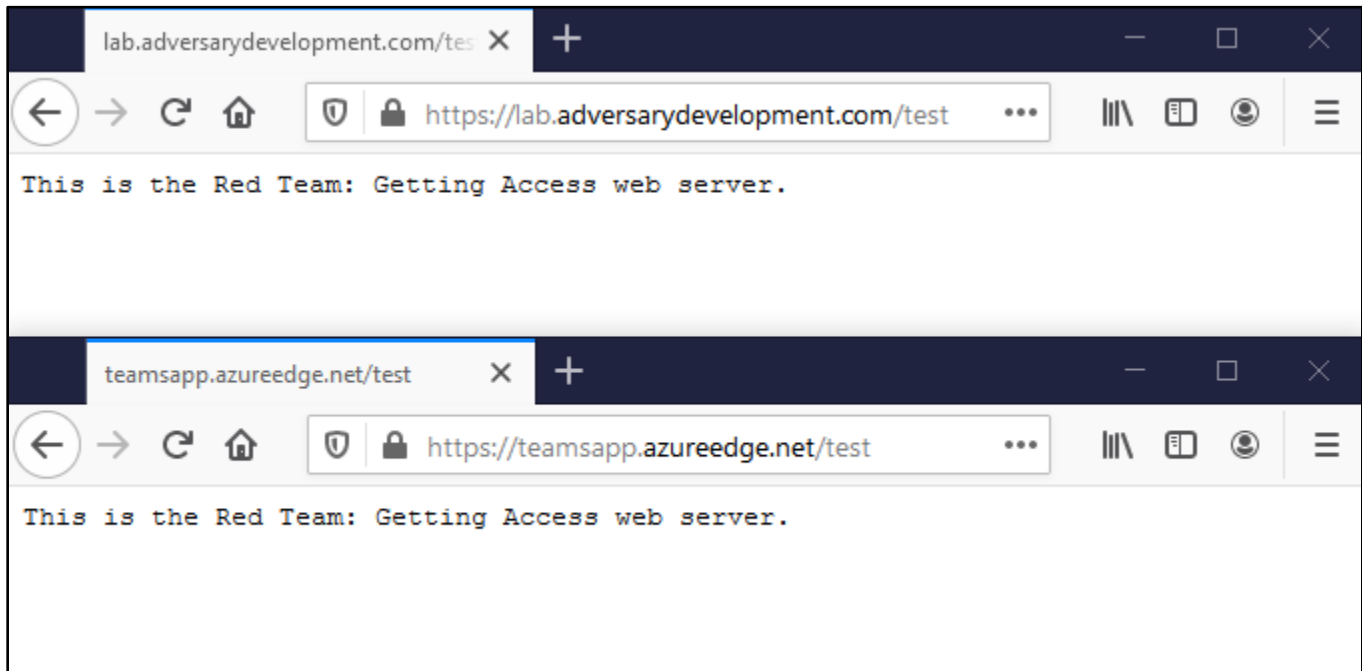
19. It may take a few minutes for your endpoint to become fully configured in Azure. In the next section, you can experiment with the CDN redirector I setup for the exercise (teamsapp.azureedge.net) if your endpoint isn't ready yet.

2. Observing the effects of the CDN redirector

1. In a web browser, visit the CDN endpoint you just created and compare it with the web server it redirects to. You should see the same content on both servers, since your request to the CDN endpoint is just being relayed to your actual server. If your CDN endpoint isn't ready yet, you can follow along using my CDN endpoint and origin server URLs listed below.

Origin server: <https://lab.adversarydevelopment.com/test>

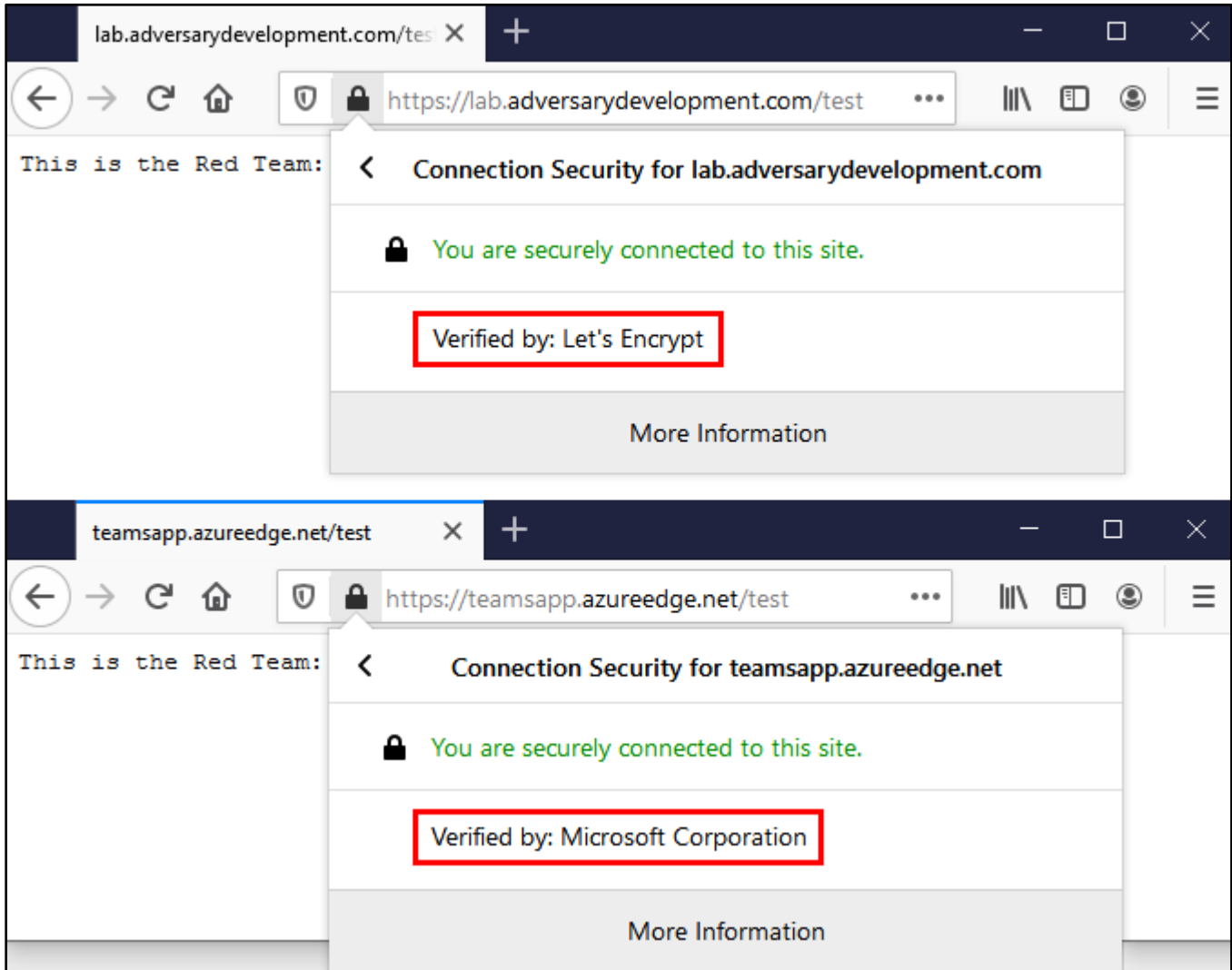
CDN endpoint (redirector): <https://teamsapp.azureedge.net/test>



Comparison of Content Served by the CDN Redirector and the Real Web Server

2. While you still have the two different URLs open in a web browser, inspect each web server's encryption certificate. Note that while your origin server's certificate may be signed by a different authority (mine is signed

by Let's Encrypt), the CDN endpoint's certificate is signed by Microsoft, which could make it look more trustworthy - especially if you're impersonating a Microsoft service or application, like this one is.



Comparison of Encryption Certificates

- Next, observe the DNS records associated with the CDN endpoint and origin server. If you run the following commands in your Kali Linux VM, you may notice that the IP address of your original server is not disclosed by the CDN endpoint - helping to hide the origin server's real location on the Internet.

```
dig +short lab.adversarydevelopment.com
```

```
dig +short teamsapp.azureedge.net
```

```
unknown@*:~$ dig +short lab.adversarydevelopment.com
164.90.139.5
unknown@*:~$ dig +short teamsapp.azureedge.net
teamsapp.afd.azureedge.net.
star-azureedge-prod.trafficmanager.net.
dual.t-0009.t-msedge.net.
t-0009.t-msedge.net.
Edge-Prod-ATAr3a.ctrl.t-0009.t-msedge.net.
standard.t-0009.t-msedge.net.
13.107.246.19
13.107.213.19
```

Comparison of DNS Resolution Results

- Finally, examine the reputation of both domains online, using a service such as Symantec WebPulse (URL below). Note that although you haven't taken any steps to establish your CDN endpoint's reputation, it is already categorized based on the reputation of the parent domain. By contrast, the origin server has not been categorized - making it more likely to be blocked by a filtering web proxy.

```
https://sitereview.bluecoat.com/
```

WebPulse Site Review Request


[Check another URL](#)

URL submitted:

<https://teamsapp.azureedge.net:443/> 

Current categorization:

[Technology/Internet](#)

Last Time Rated/Reviewed: > 7 days 

Categorization Inherited from the CDN Parent Domain

WebPulse Site Review Request

[Check another URL](#)

URL submitted:

<https://lab.adversarydevelopment.com:443/>

This URL has not yet been rated

Since this URL has not yet been rated, please fill out the form below so we can add it to our database.

Lack of Categorization on the Origin Server by Default