Utilizando o New-AzFrontDoorCdnRuleSslProtocolConditionObject no PowerShell

Title: Using New-AzFrontDoorCdnRuleSslProtocolConditionObject in PowerShell for Windows

Introduction: In this article, we will explore the usage of the New-AzFrontDoorCdnRuleSslProtocolConditionObject cmdlet in PowerShell for Windows. This cmdlet allows us to define SSL protocol conditions for Azure Front Door CDN rules, providing more control over the security of our applications and websites.

Importance: SSL protocol conditions are essential for enforcing secure communication between clients and servers. By utilizing the New-AzFrontDoorCdnRuleSslProtocolConditionObject, we can define specific SSL protocols that are allowed or denied for incoming requests. This ensures that only secure and approved protocols are utilized, reducing the risk of vulnerabilities and unauthorized access.

Examples: To illustrate the usage of the New-AzFrontDoorCdnRuleSslProtocolConditionObject, let's consider a scenario where we want to allow only TLS 1.2 and TLS 1.3 protocols for our Front Door CDN rules.

1. Install the Azure PowerShell module if not already installed:

```
Install-Module -Name Az -AllowClobber -Scope CurrentUser
```

2. Connect to your Azure account:

```
Connect-AzAccount
```

3. Define the SSL protocol condition object:

```
$sslProtocolCondition = New-AzFrontDoorCdnRuleSslProtocolConditionObj
ect -MinProtocolVersion TLSv1_2 -MaxProtocolVersion TLSv1_3
```

4. Create a new Front Door CDN rule using the SSL protocol condition:

```
$rule = New-AzFrontDoorCdnRule -Name "MyCDNRule" -FrontDoorName "MyFr
ontDoor" -MatchVariable RequestProtocol -Operator Equal -MatchValue "
https" -ActionType Redirect -RedirectType Moved -RedirectProtocol Htt
ps -RedirectCustomPath "/redirected" -RedirectQueryString False -Enab
ledState Enabled -SslProtocolCondition $sslProtocolCondition
```

In the above example, we create a new Front Door CDN rule named "MyCDNRule" and specify the SSL protocol condition using the New-AzFrontDoorCdnRuleSslProtocolConditionObject. We set the minimum protocol version to TLS 1.2 and the maximum protocol version to TLS 1.3. This ensures

that only requests using these protocols will be allowed for the rule.

Alternatives: If you are not using the Windows environment, you can still utilize Azure Front Door CDN and define SSL protocol conditions using other methods. One alternative is to use the Azure CLI, which provides similar functionalities to PowerShell. The equivalent command in Azure CLI for creating a Front Door CDN rule with SSL protocol conditions would be:

```
az network front-door cdn-rule create --name MyCDNRule --front-door-name M
yFrontDoor --match-variable RequestProtocol --operator Equal --match-value
 "https" --action-type Redirect --redirect-type Moved --redirect-protocol
Https --redirect-custom-path "/redirected" --redirect-query-string False -
-enabled-state Enabled --ssl-protocol-versions "TLSv1_2,TLSv1_3"
```

Conclusion: By utilizing the New-AzFrontDoorCdnRuleSslProtocolConditionObject in PowerShell for Windows, we can easily define SSL protocol conditions for Azure Front Door CDN rules. This enables us to enforce secure communication by allowing or denying specific SSL protocols. Understanding and implementing SSL protocol conditions is crucial for maintaining the security and integrity of our applications and websites.