1) Cross-Origin Resource Sharing (CORS)

Access-Control-Allow-Headers

Content-Type, soapaction

Access-Control-Allow-Methods

GET, POST, PUT, DELETE, OPTIONS

Access-Control-Allow-Origin

\*

2) Insecure Content Security Policy (CSP)/X-Frame-Options

X-Content-Type-Options

nosniff

X-Frame-Options

SAMEORIGIN

X-Xss-Protection

1; mode=block

3) Browser Cache Enabled

Cache-Control

no-cache, no-store

4) Missing HSTS Header

Strict-Transport-Security

max-age=31536000; includeSubDomains; preload

Content Security Policy (CSP) (Working Value)

default-src 'self'; script-src 'unsafe-inline' 'self' https://www.google-analytics.com/ https://www.googletagmanager.com/; style-src 'self' 'unsafe-inline' https://fonts.googleapis.com/; font-src 'self' https://fonts.gstatic.com/; frame-src 'self'; connect-src 'self' https://www.google-analytics.com/ https://stats.g.doubleclick.net/; img-src 'self' https://www.google.co.in/ https://www.google.com/ ;

HTTP PUBLIC KEY PINNING#####

Name : Public-Key-Pins

Value : pin-sha256='MHJYVThihUrJcxW6wcqyOISTXIsInsdj3xK8QrZbHec='; includeSubdomains; max-age=2592000

X-POWERED-BY####

Name : X-Powered-By

Value : Hello You

######X-ASPNET-VERSION####

<system.web>

<httpRuntime enableVersionHeader="false" />

</system.web>

Application Error Message.

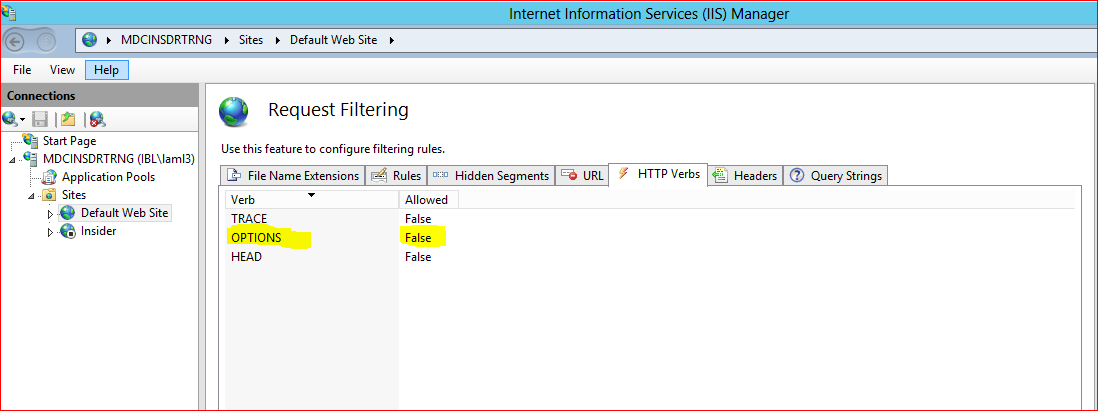
Error Page 🡺 Detailed errors for local request and custom error pages for remote request.

Set the default page and give the error massage path



HTTP Verb Tampering

Request Filtering🡺 HTTP Verb🡺Options false



Weak TLS CBC cipher Detected.

Run🡺regedit🡺computer\HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Cryptography\Configuration\Local|SSL\00010002

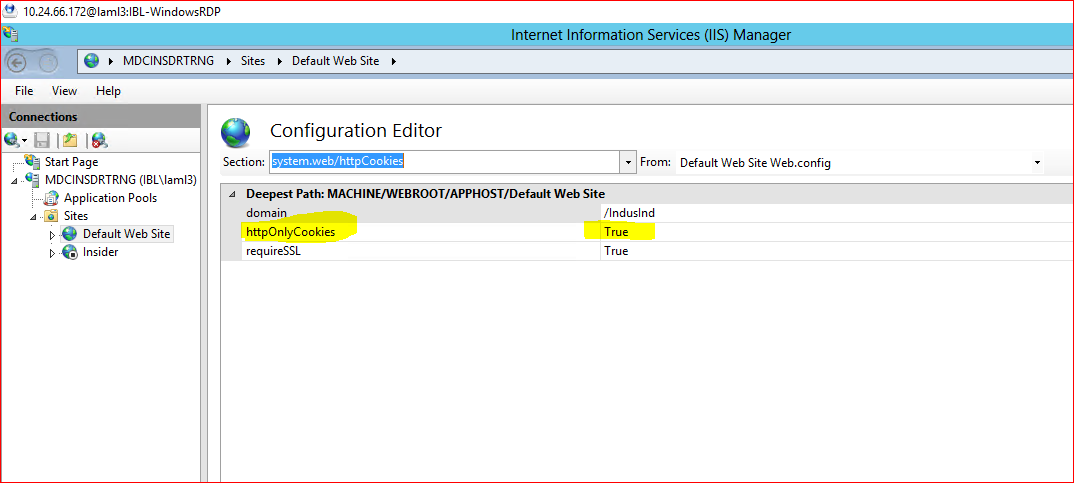
Removed below selected CBC cipher🡺 than click Ok.



Missing Secure Flag from SSL Cookie

Missing Secure Flag From Cookie Header

Go to Configuration Editor🡺 httpOnlyCookies: True



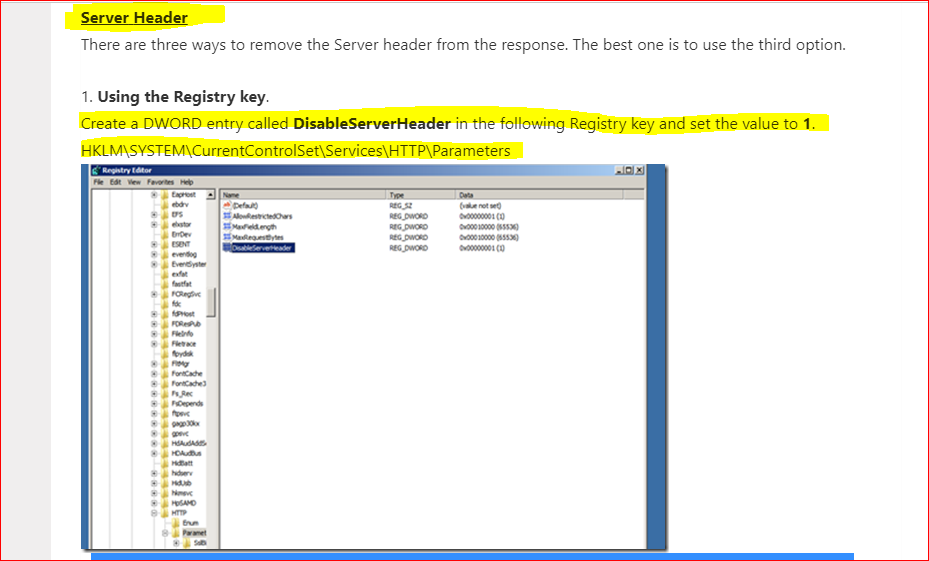
<system.web

<httpCookies httpOnlyCookies="true" requireSSL="true" />

</system.web>

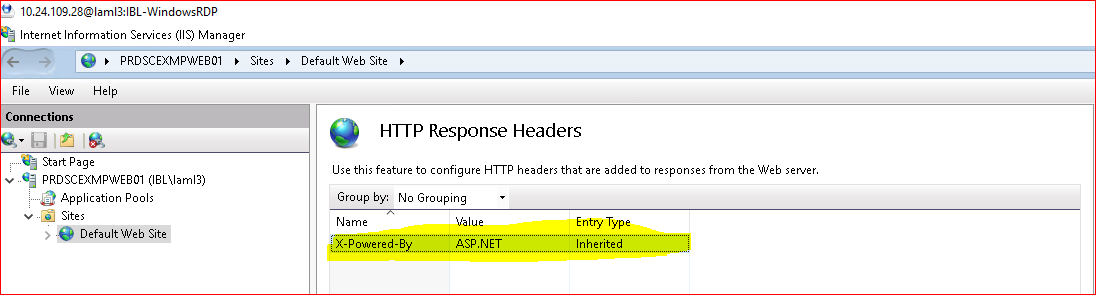
Web Server Information Disclosure

**Customers are advised to modify the HTTP response header of the target application to not disclose detailed information about the underlying web server. Server implementers are encouraged to make this field a configurable option**



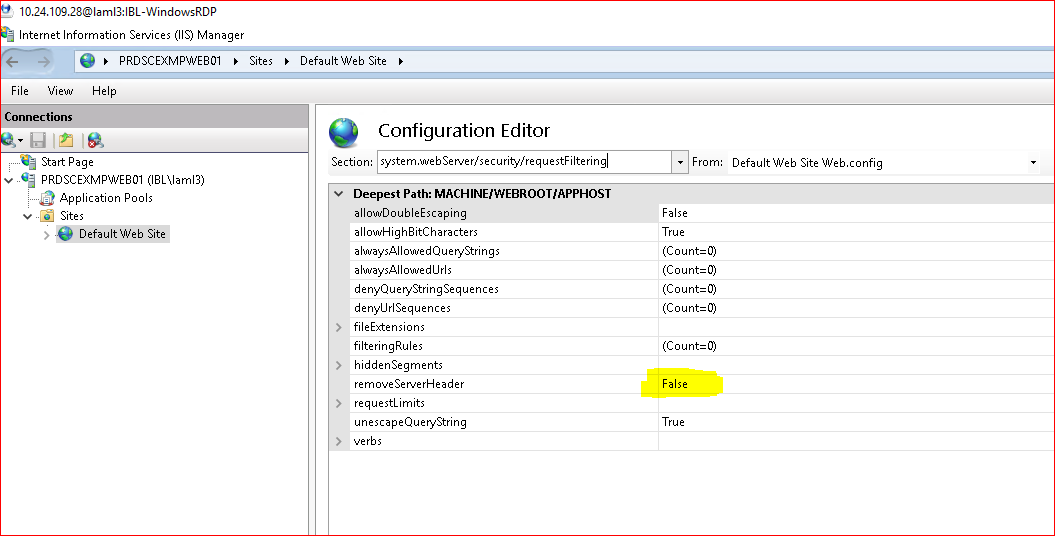
Removed X-Powered-By

Go🡺 to HTTP Response Headers🡺 removed below mentioned X-Powered-By Header.



Change to True.

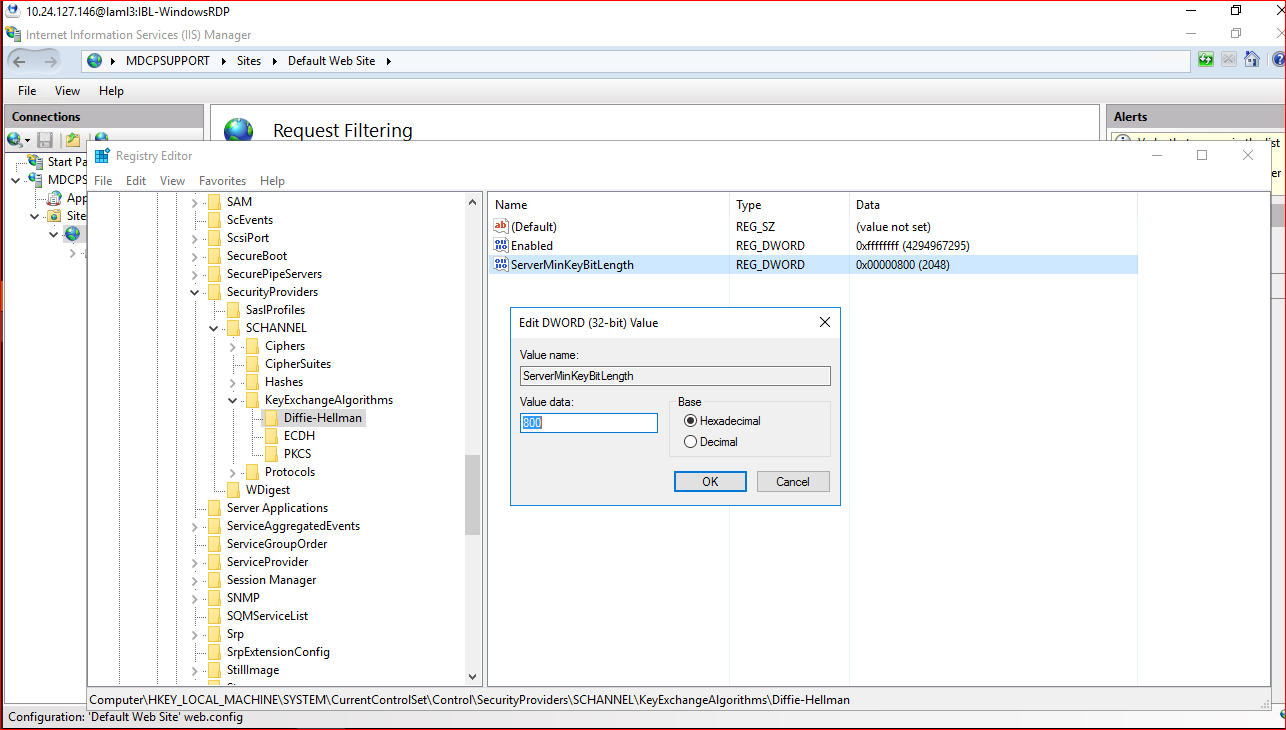
Go to 🡺 Configuration Editor🡺 RemoveServerHearder🡺Change to **True.**



**Diffie-Hellman Ephemeral Key Exchange DoS Vulnerability (SSL/TLS, D(HE)ater)**

[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\KeyExchangeAlgorithms\Diffie-Hellman]

Set to **ServerMiniketBitLenght** 🡺 value to **800** than click ok.



**Diffie-Hellman Ephemeral Key Exchange DoS Vulnerability (SSL/TLS, D(HE)ater**

Remove this cipher🡺 TLS\_DHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256

Remove this cipher🡺 TLS\_DHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384

**SSL/TLS: Report Vulnerable Cipher Suites for HTTPS**

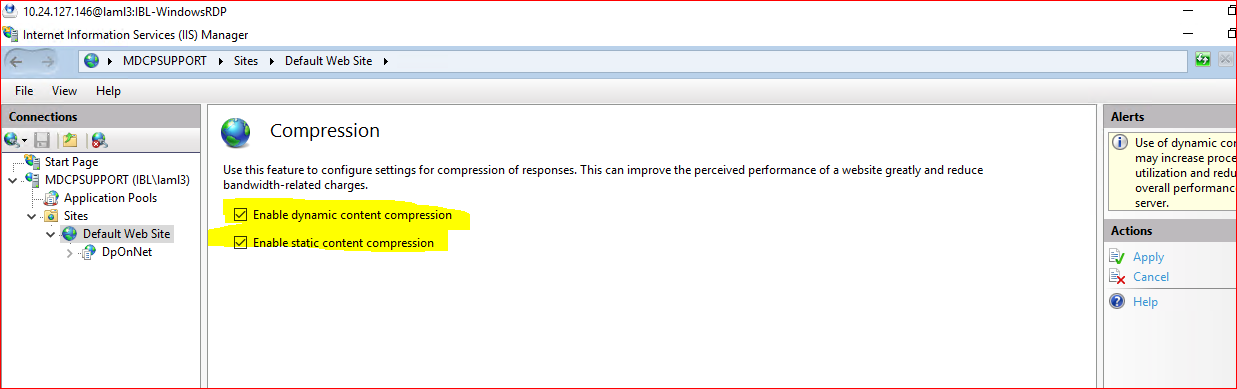
TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA

SSL/TLS: BREACH attack against HTTP compression.

Go to 🡺**Compression**

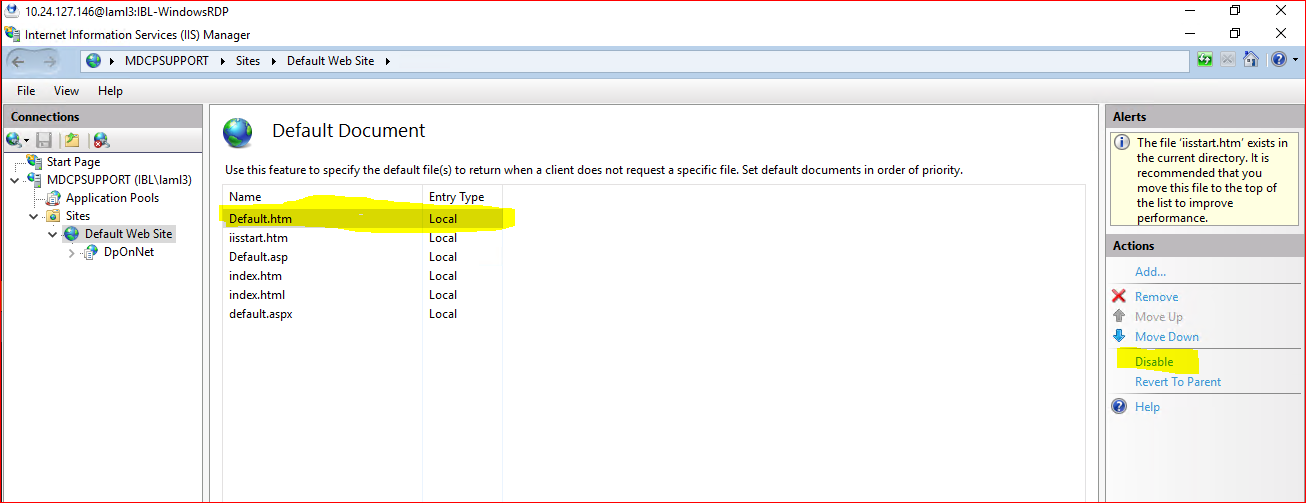
Check Enable dynamic content Compression.

Check Enable static Content Compression.



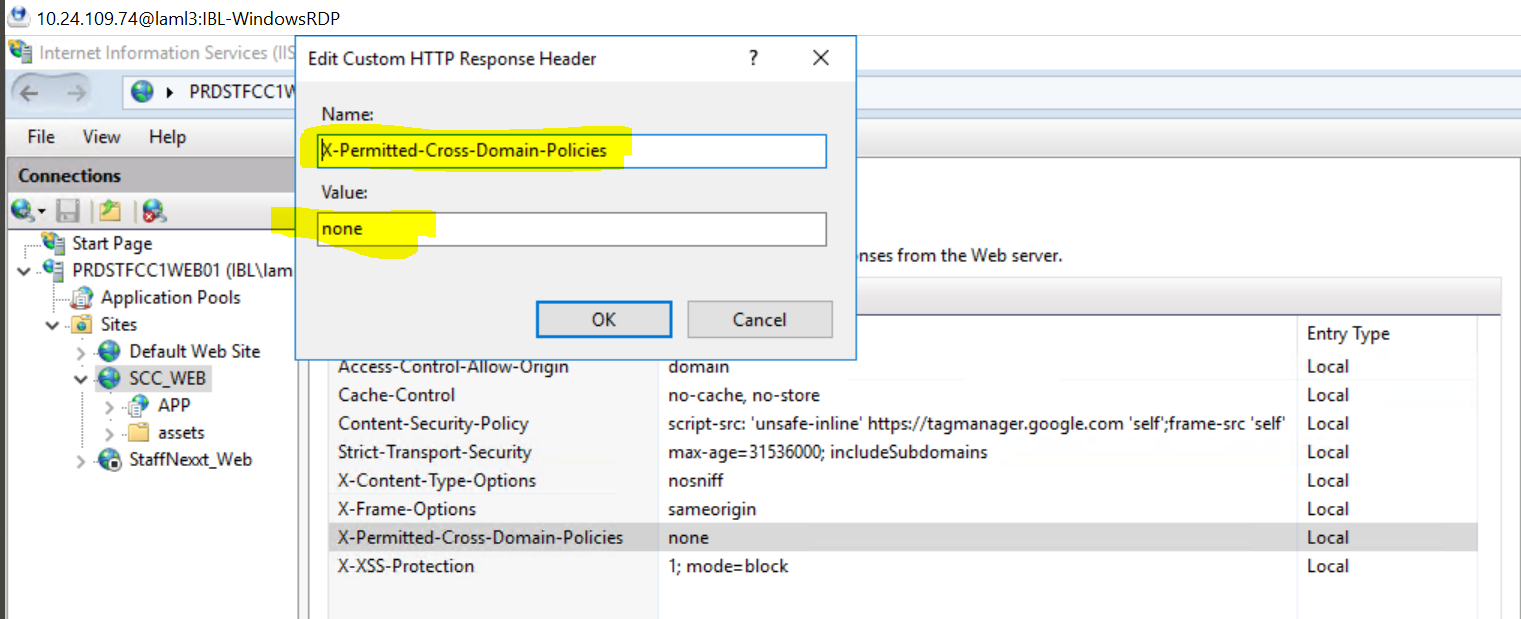
**Web Server Default Web Page Detected.**

Go🡺 to Default Document and Click the Default.htm page and disable it from the Actions pane appears in the right side.

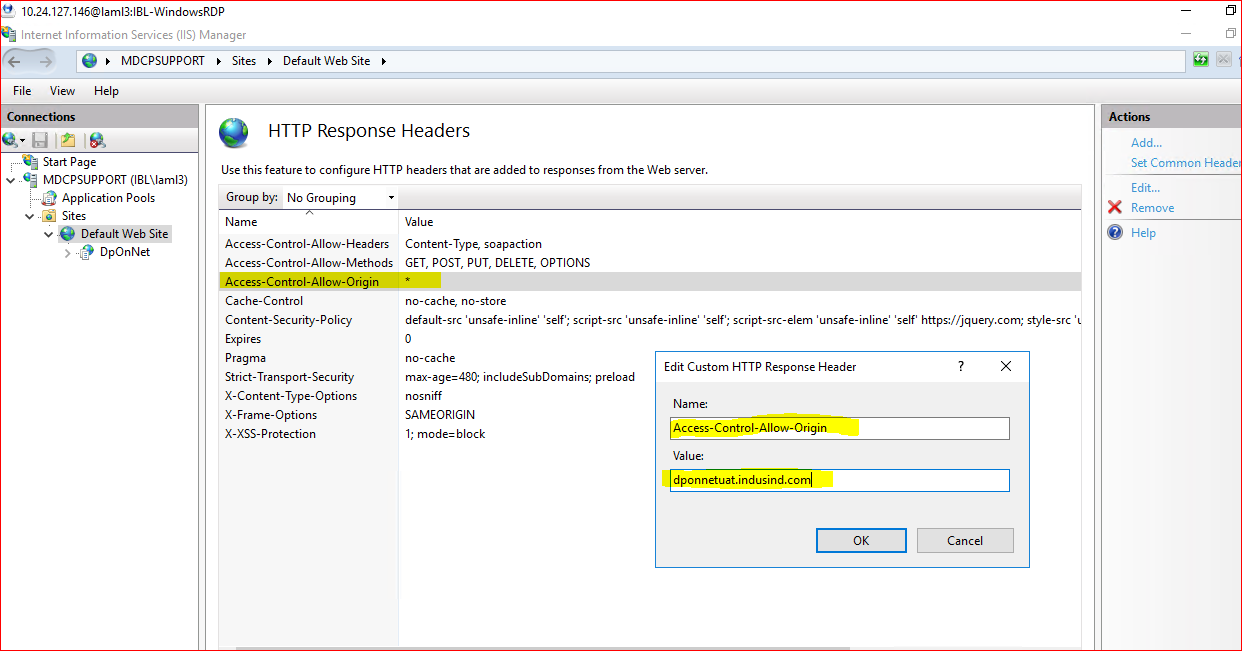


**Unset/Insecure X-Permitted-Cross-Domain-Policies Header.**

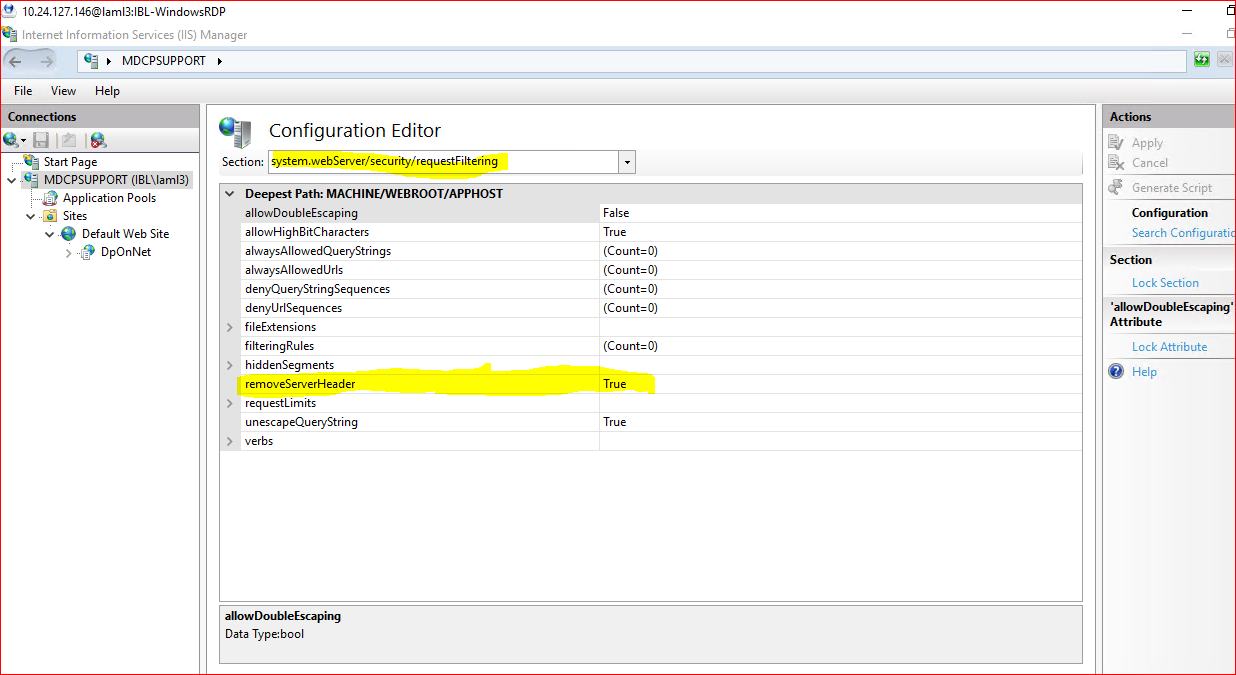
**Go🡺 to HHTP Response Headers🡺** right and side action pane click on Add Button**-🡺** type Name  **X-Permitted-Cross-Domain-Policies.** In the **value** Tab**🡺** Enter the host Namee.g **none;** then click on Ok button.



**Go🡺 to HHTP Response Headers🡺** right and side action pane click on Add Button**-🡺** type **Name Access-Control-Allow-Origin.** In the **value** Tab**🡺** Enter the host Namee.g **abc.com** then click on Ok button.

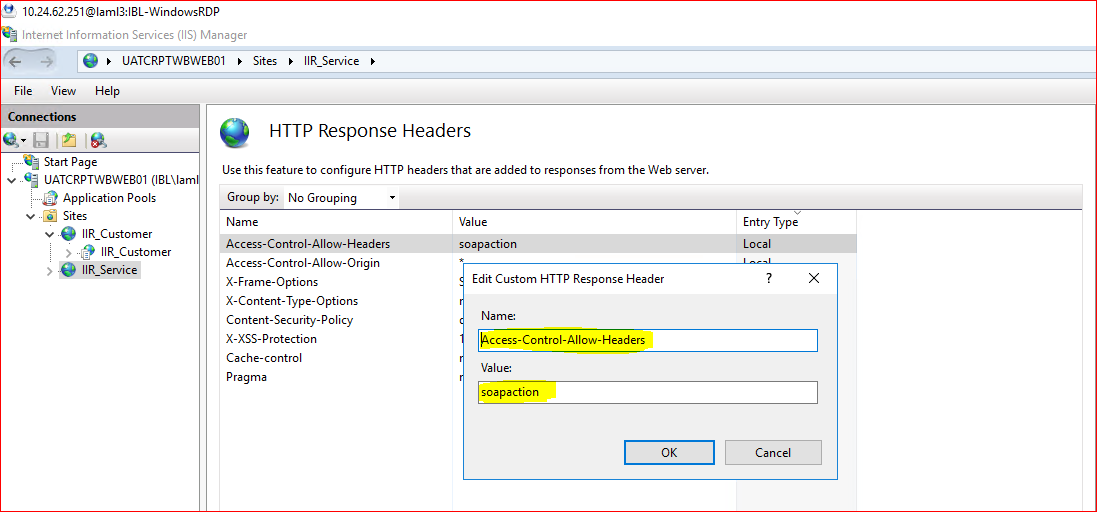


**The target application discloses the Web Server software via the "Server:" token sent in HTTP response header.   
  
Go🡺 to Configuration Editor🡺 System.webServer🡺Security🡺requestFiltering>>removerServerHeader🡺True**

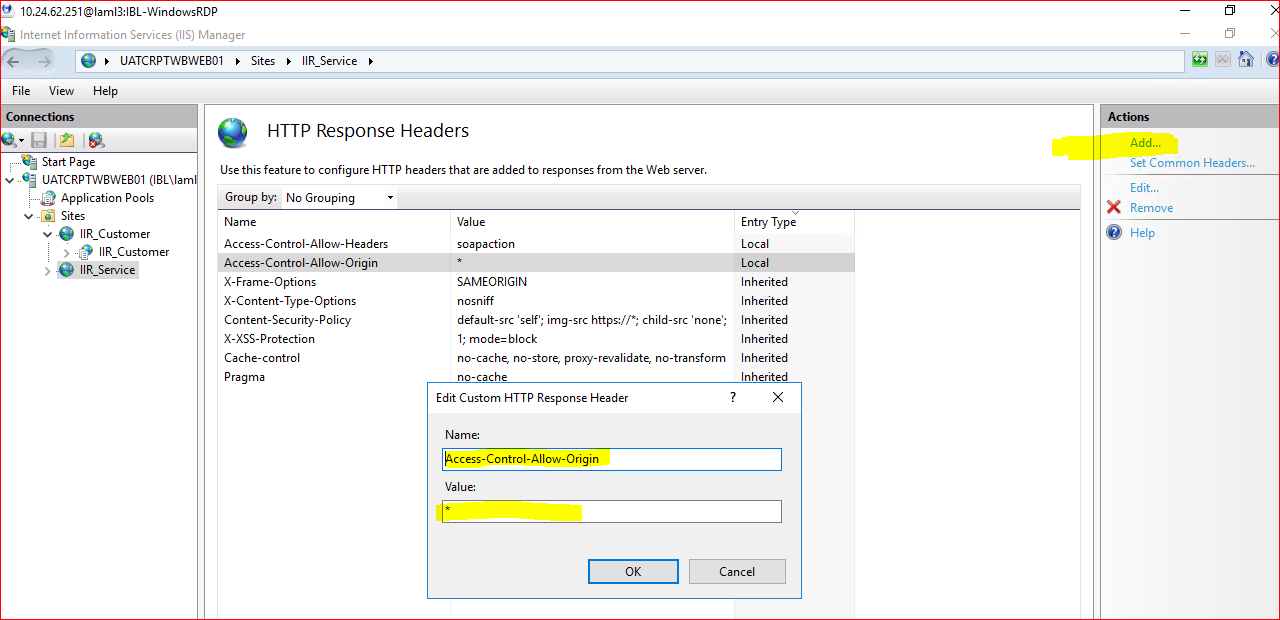


**Cross-Origin Resource Sharing (CORS)**

Go to 🡺 HTTP Response Headers🡺 right side action pane click on add button and enter the below value

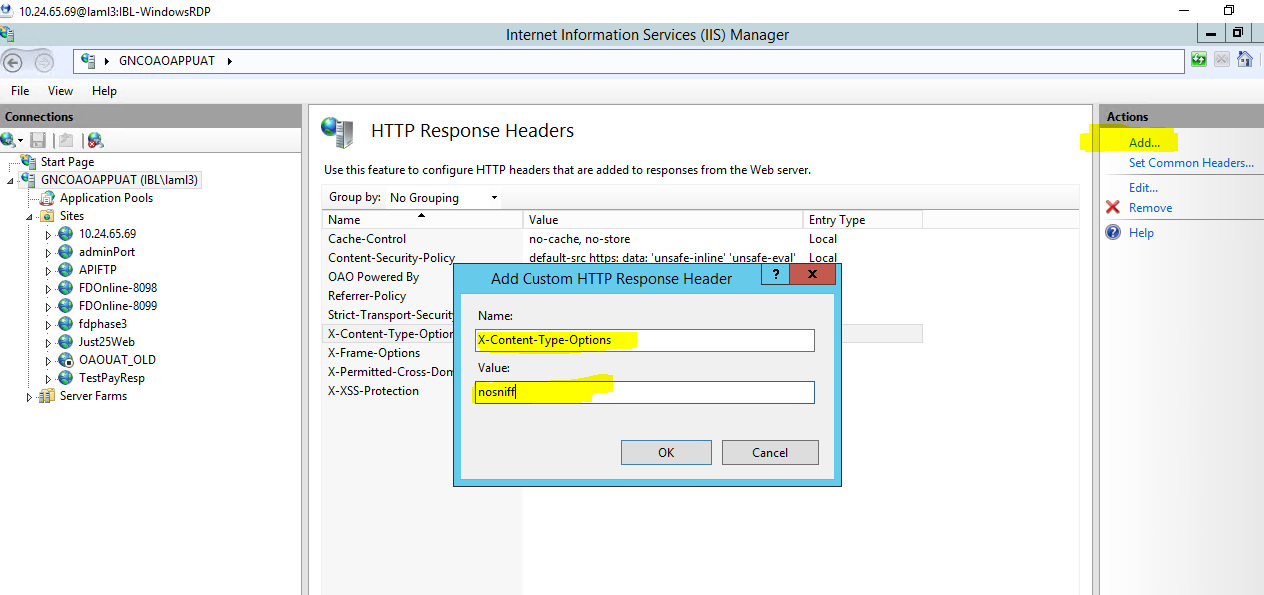


Go to 🡺 HTTP Response Headers🡺 right side action pane click on add button and enter the below value



**Web Server Content Sniffing Enabled.**

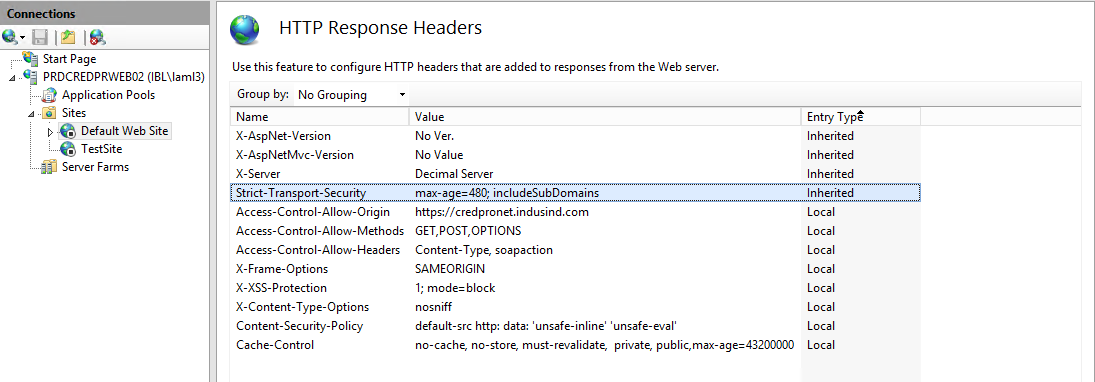
Go to 🡺 HTTP Response Headers🡺 right side action pane click on add button and enter the below value



**Internal IP Address Disclosure.**

Application should only reveal domain names.

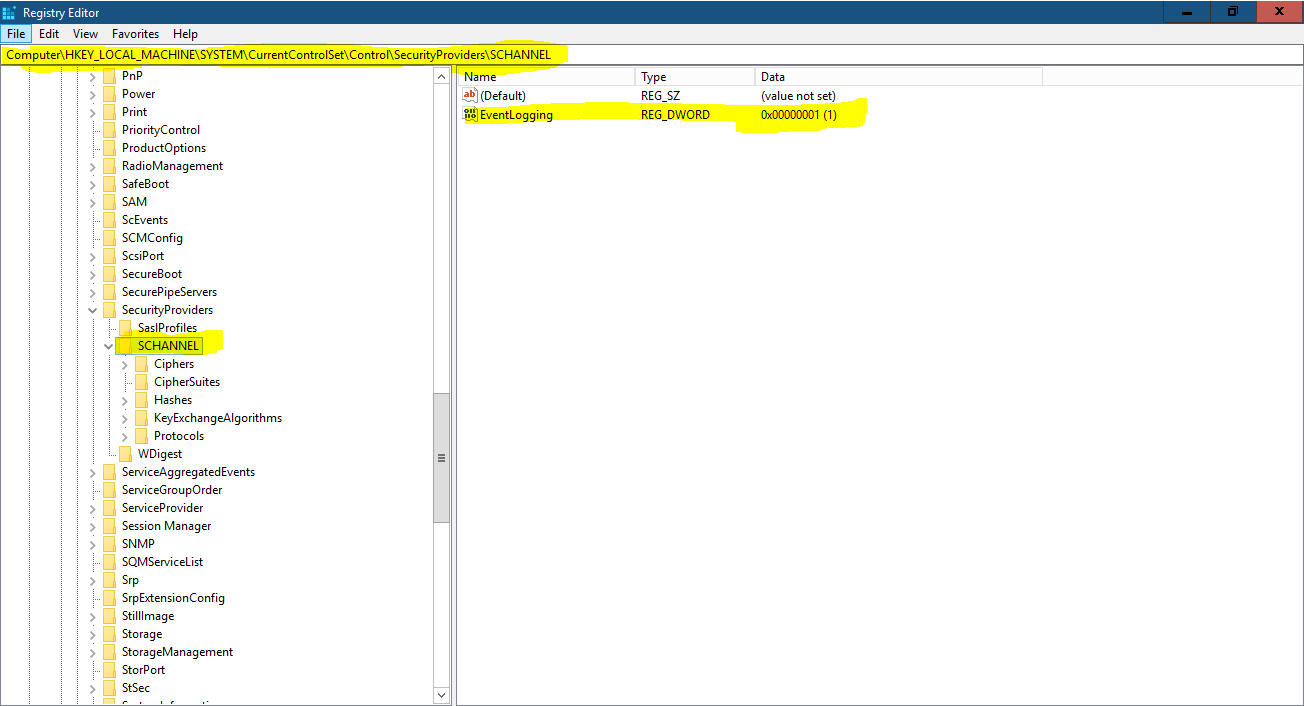
**Server Headers.**



**Weak ciphers supported**

Expand System 🡺 Current Control Set 🡺 Control 🡺 Security Providers 🡺 SCHANNEL.

🡺EventLoging value should be 0.



**Missing HTTP Security Headers**

It is recommended to implement below headers:

• Strict Transport Security

• Cache-control

**HTTP STRICT TRANSPORT SECURITY#####**

**Name : Strict-Transport-Security**

**Value : max-age=31536000; includeSubdomains**

**Browser Cache Enabled**

**Name: Cache-Control**

**Value : no-cache, no-store, must-revalidate, private, public,max-age=43200000**

**Name : Referrer-Policy**

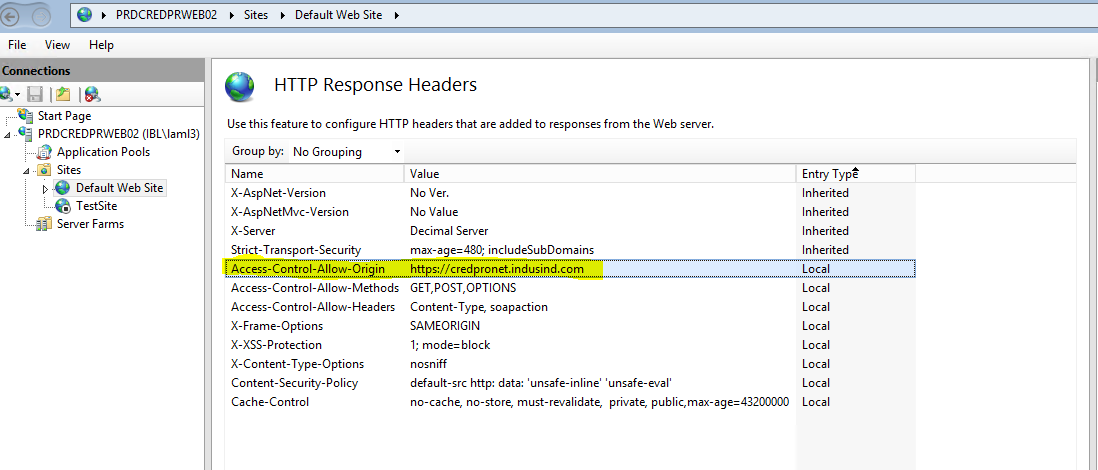
**Value :** strict-origin-when-cross-origin



**The application is vulnerable for cross origin resource sharing**

**name="Access-Control-Allow-Origin"**

**value=" abc.com”**



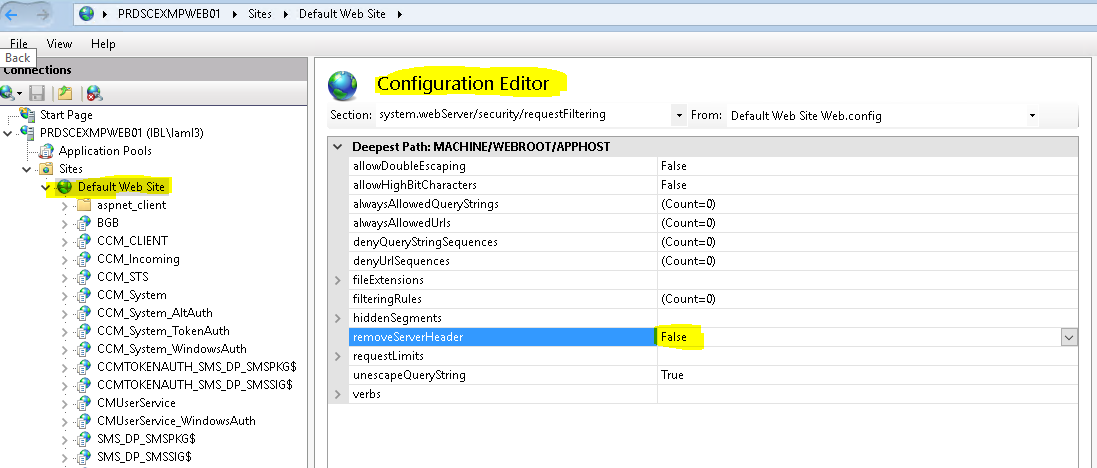
**An attacker can fingerprint the web server from HTTP responses**

**Go to 🡺 Web.config File 🡺 In the Security tab🡺 add**

**<requestFiltering removeServerHeader="true" />**

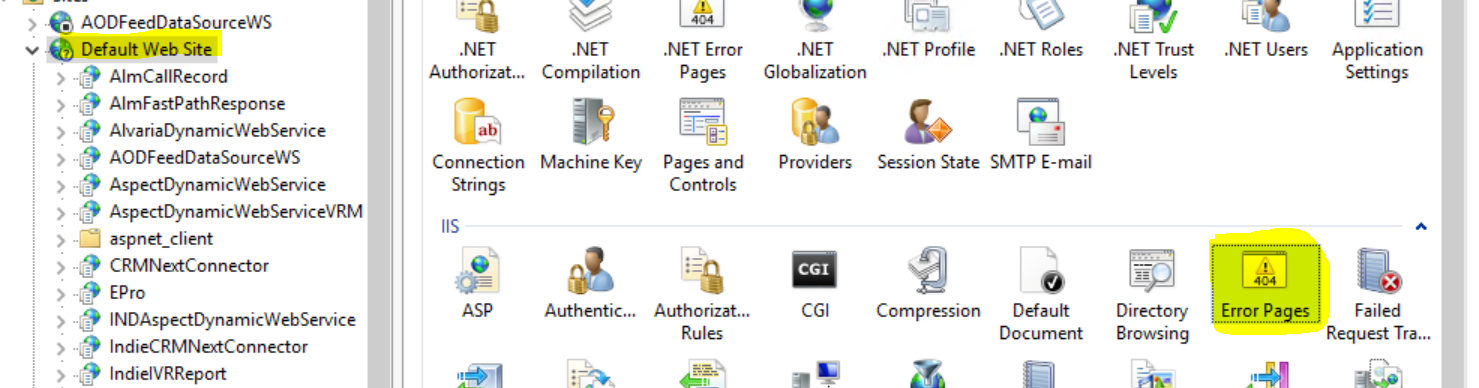


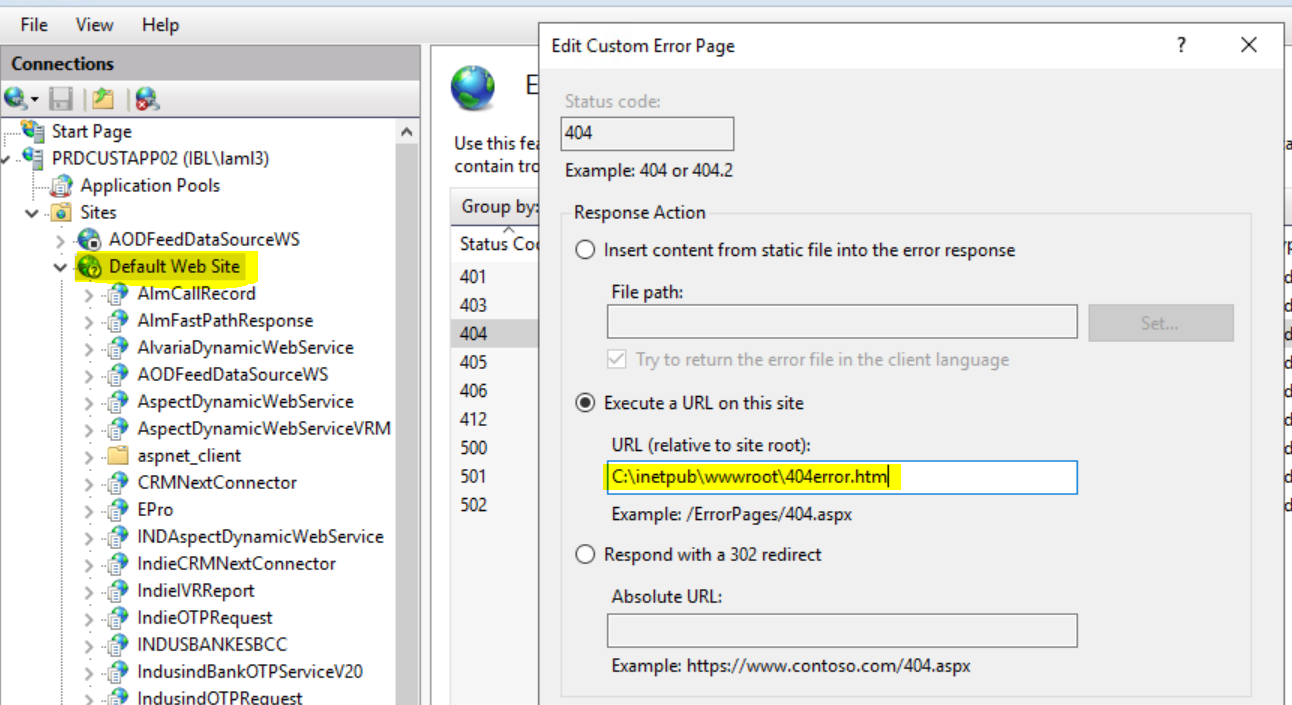
**Go to🡺 Configuration Editor🡺 removeServerHeader🡺 set to true**



**An attacker might be able to use sensitive information revealed by error messages to launch further attacks**

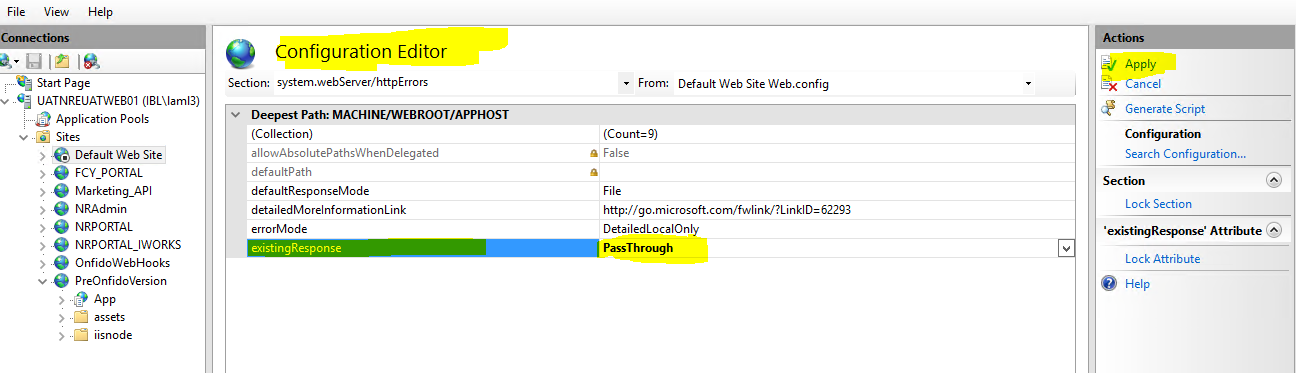
**Go to 🡺 IIS 🡺 Default Web Site**and open up the**Error Pages🡺** Double click on the 404 status code, choose the Execute a URL to this site option and enter in the path and click "OK"





**Internal path disclosure.**

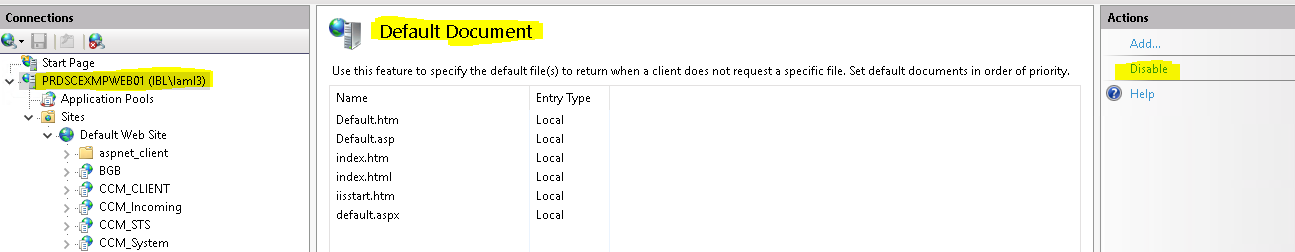
Go To Configuration Editor 🡺 system.webServer/httpErrors🡺 Change existingResponse to PassThrough🡺 click on apply.



**IIS default page is available**

🡺Click the server name🡺Double click on Default Document

🡺On the right side, click “Disable”



**Disable OPTIONS method.**

**HTTP OPTIONS Method Enabled**

To disable options in IIS, disallow the OPTIONS verb out of the rulings of HTTP Verb Request Filtering in IIS. To do this, follow these steps:

**Step 1**: Click to Open IIS Manager.

**Step 2:**Click on the name of the appliance to set it up globally. Alternatively, alter the particular website that you are setting this up for.

**Step 3:**Click on “**Request Filtering**” twice.

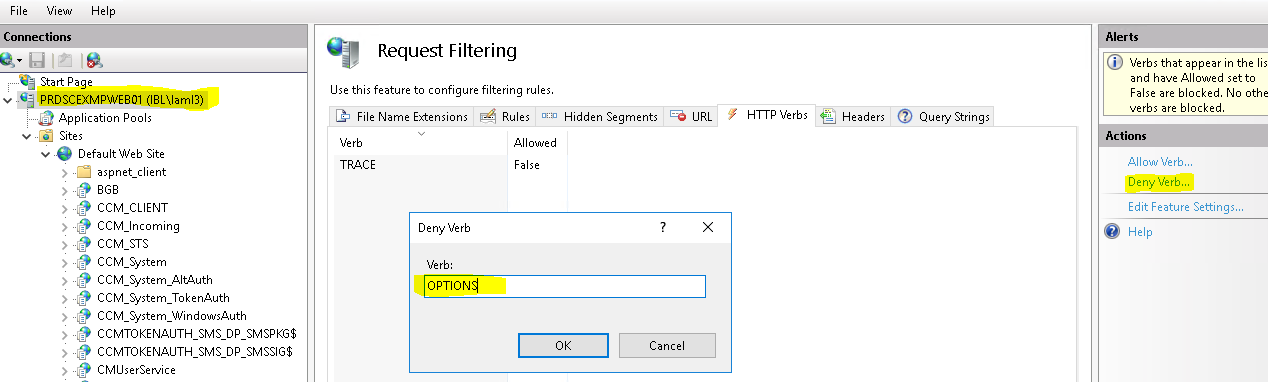
**Step 4:**Alter the button of HTTP Verbs.

**Step 5:**Visit the Actions panel and choose **“Deny Verb** “.

**Step 6:**Integrate **‘OPTIONS’** in the Verb, and click on **OK** to save the alterations you just made.

🡺Click the server name🡺Double click on Request Filtering🡺Go to HTTP Verbs tab

🡺On the right side, click Deny Verb🡺Type OPTIONS. Click OK

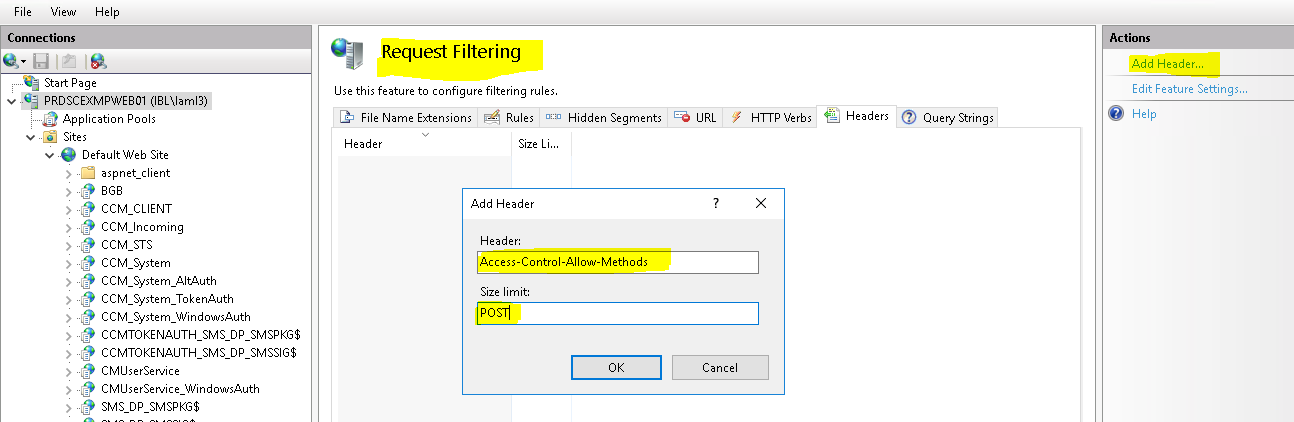


**Method Interchange for login activity.**

**Go to Request Filtering 🡺 Click on add Header🡺 Header Name**

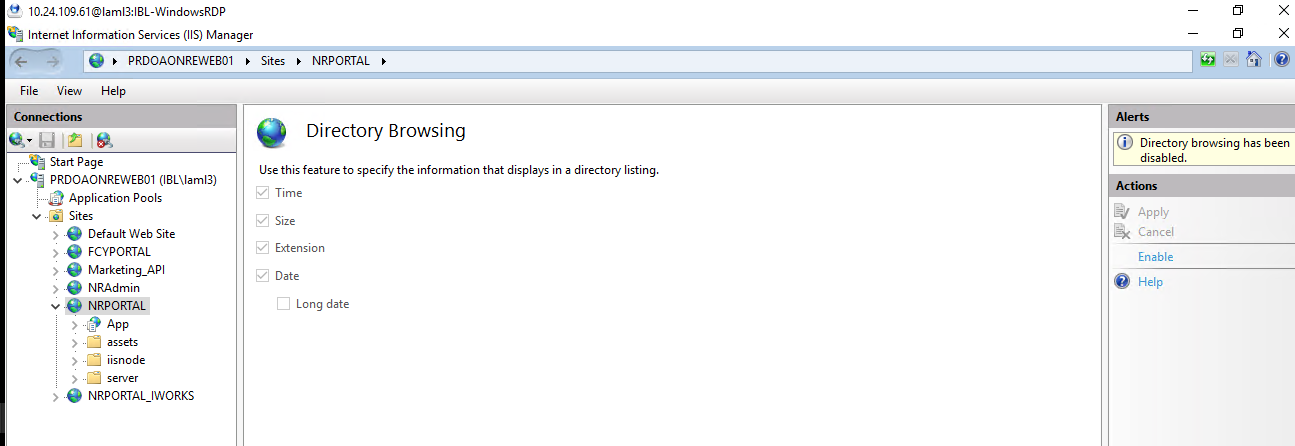
**Access-Control-Allow-Methods**

**value="GET,POST**



**Possible Sensitive Directories/Files Detected**

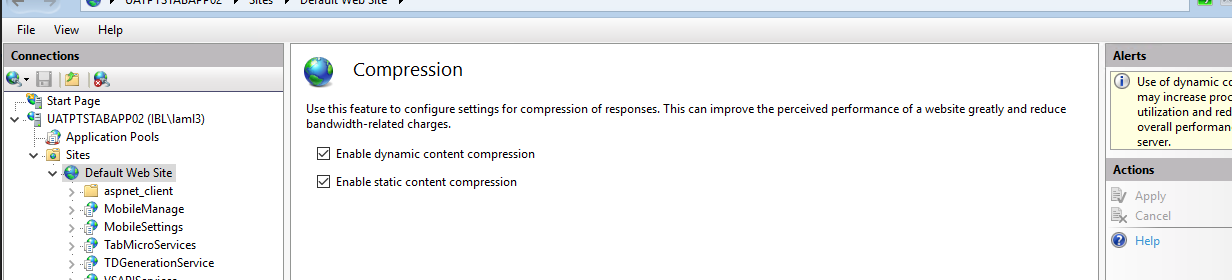
**Go to Directory Browsing🡺 Disable Directory Browsing 🡺 Click Apply**



**SSL/TLS: BREACH attack against HTTP compression.**

**Go to Compression🡪 Uncheck Enable dynamic content compression.**

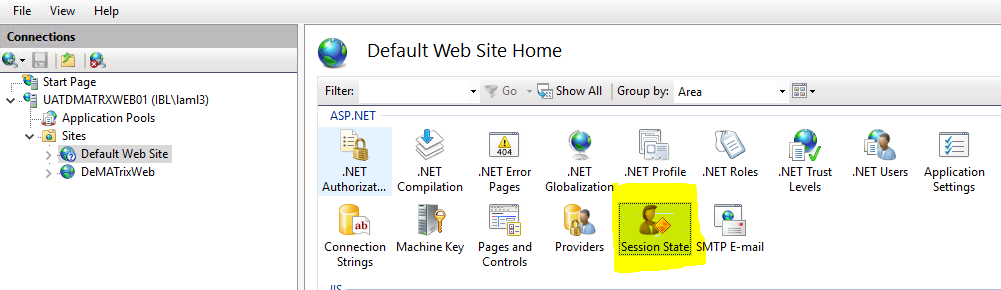
**Uncheck enable static content compression.**

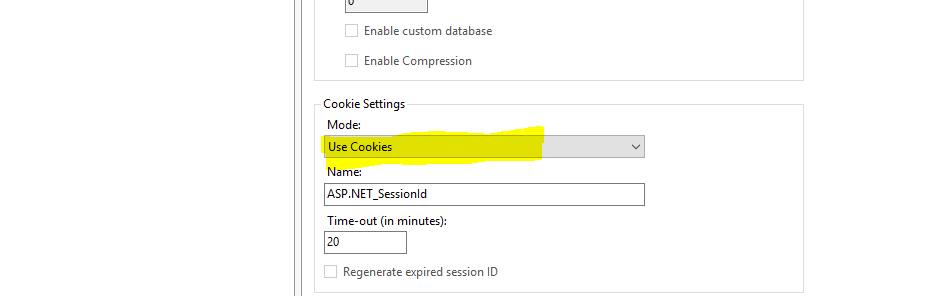


**Session cookie attributes are not set properly**

**Open IIS Manager, go to Sites | Default Web Site🡺** **double-click Sessions State.**

**Select "Use Cookies" under Mode in Cookie Settings section Click "Apply" to the right.**





**Secure HttpOnly Cookies in IIS**

**Go to system.web httpCookies🡺httpOnlyCookies🡺 Set true.**

**Go to system.web httpCookies🡺httpOnlyCookies🡺 requireSSL🡺true**

<system.web>

<httpCookies httpOnlyCookies="true" requireSSL="true" />

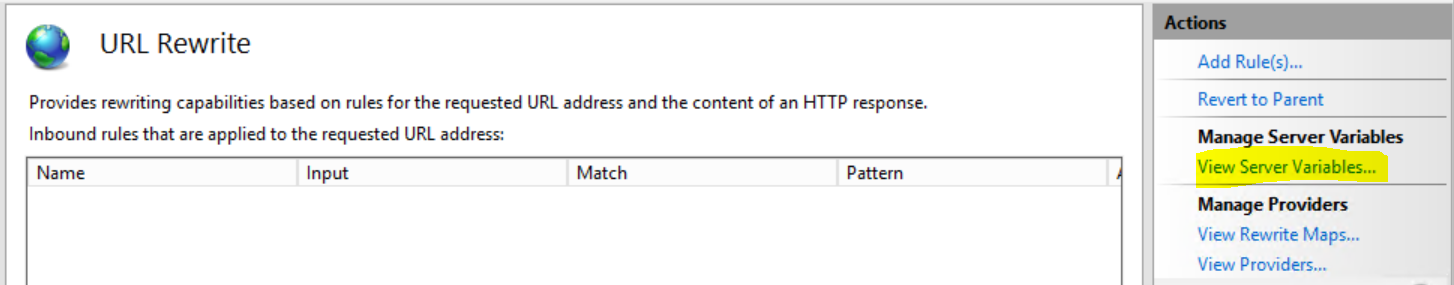


**How to remove version disclosure (IIS version 8.0,8.5) Configured (10.24.62.56)**

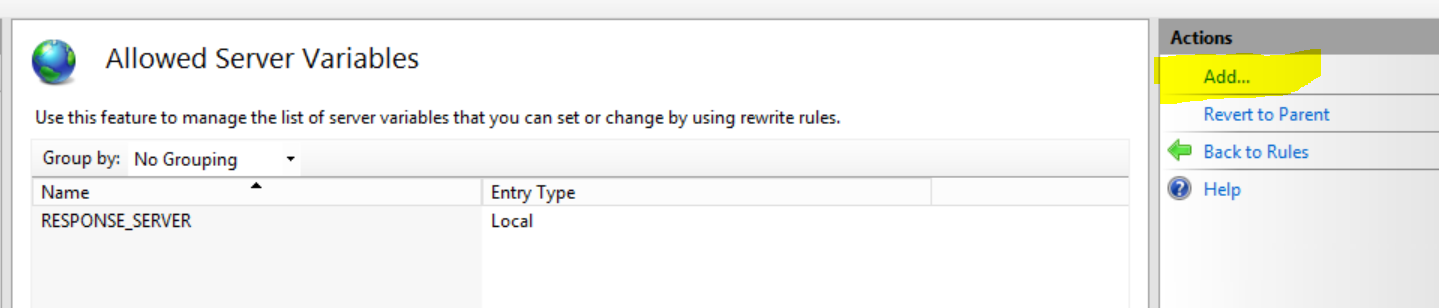
**Go to URL Rewrite🡺**



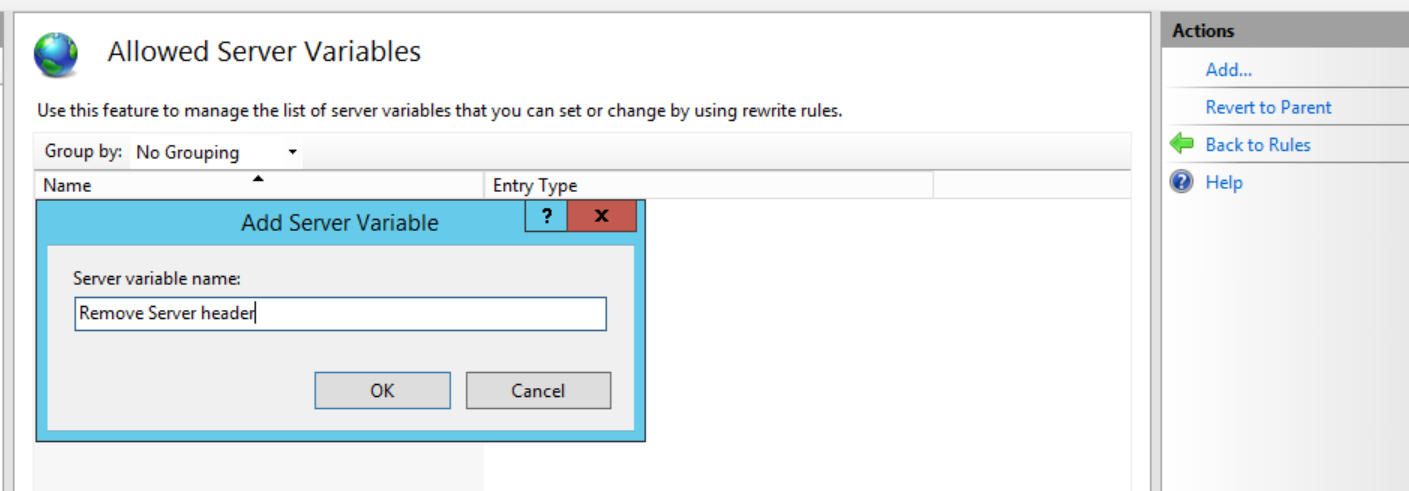
**Click on add server Variables🡺**

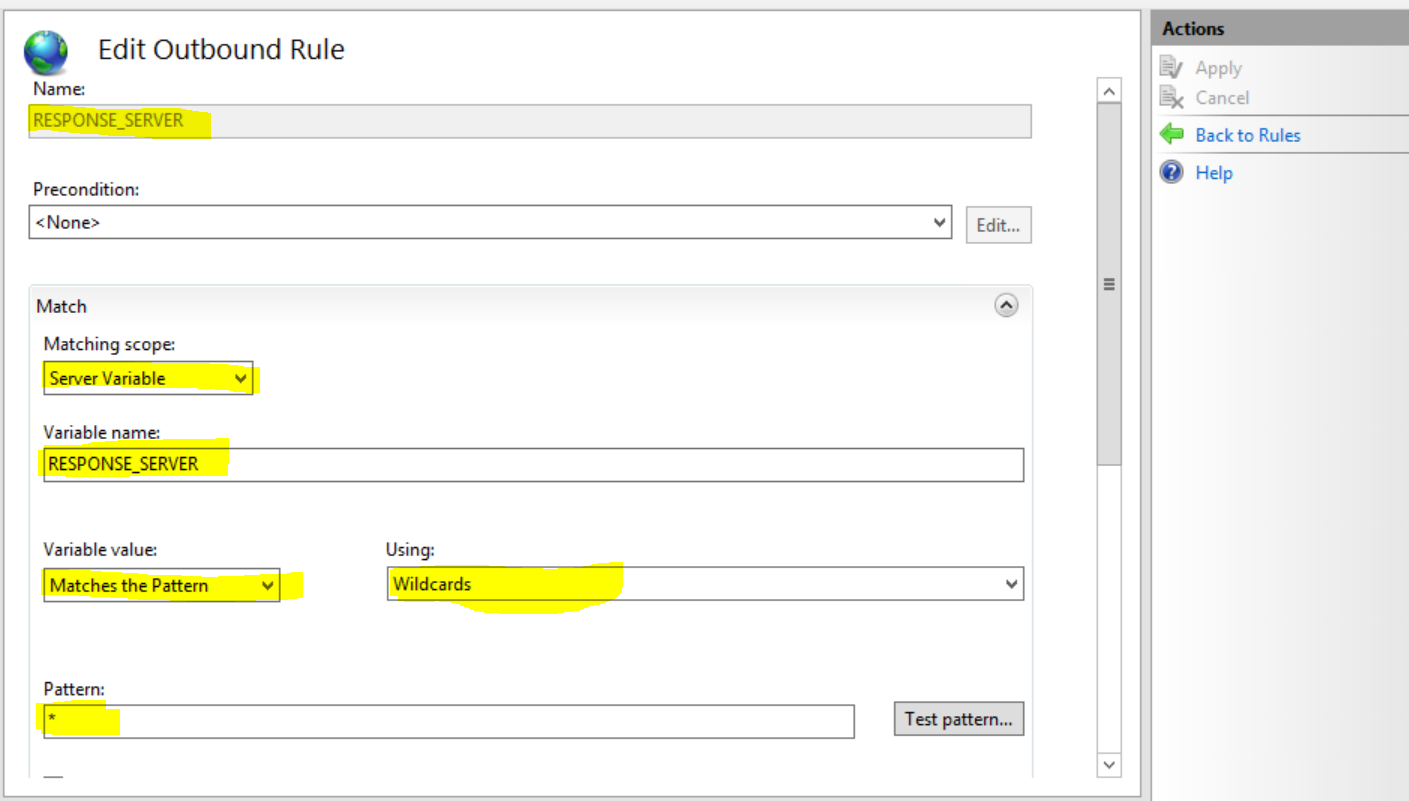


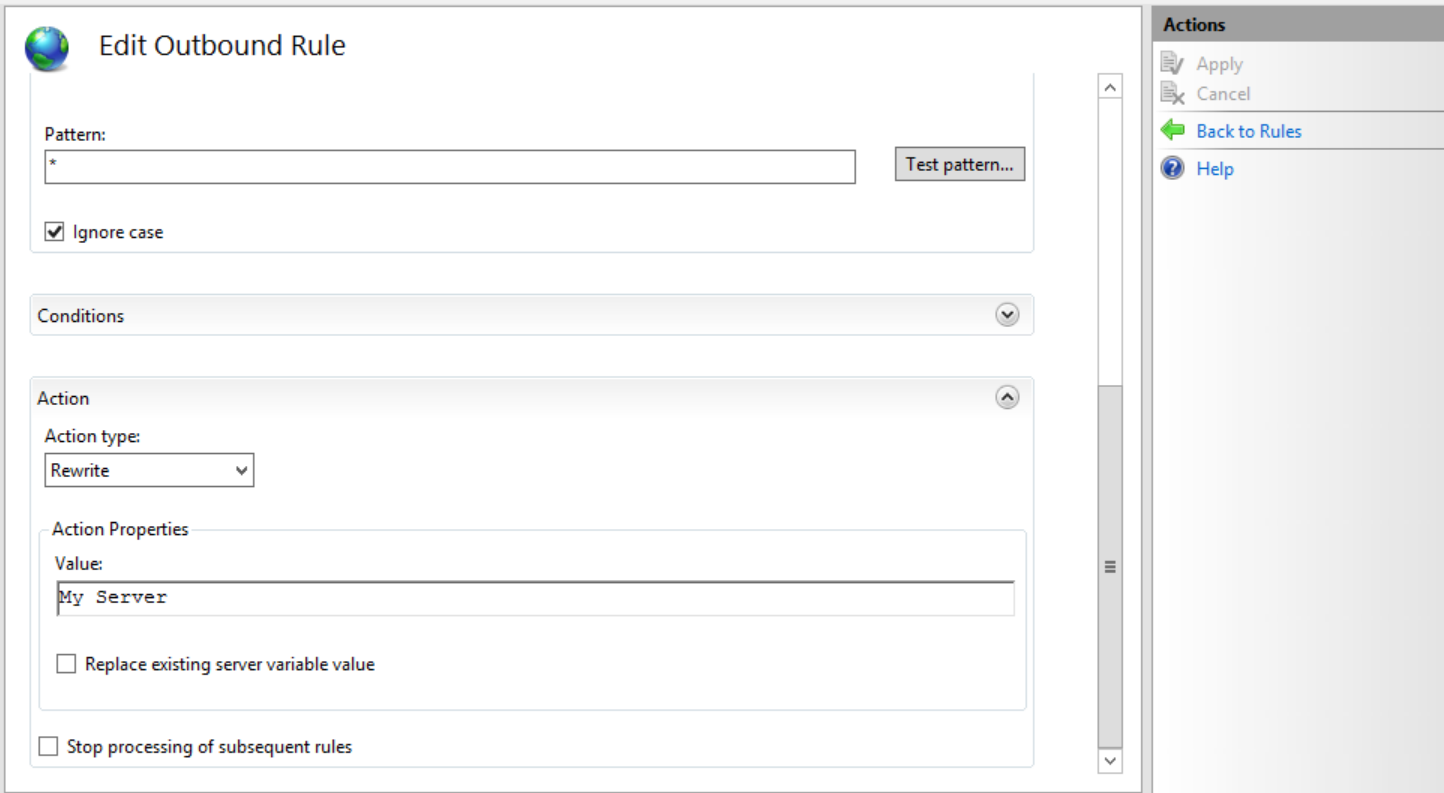
**Click on Add🡺**



**Add server Variables name 🡺 Remove server header**







**Click on Apply button.**

**OR**

**Add below entry in web.config file.**

**<rewrite>**

**<outboundRules>**

**<rule name="Remove Server Header">**

**<match serverVariable="RESPONSE\_Server" pattern=".\*" />**

**<action type="Rewrite" value="" />**

**</rule>**

**</outboundRules>**

**</rewrite>**

**<httpProtocol>**

**<customHeaders>**

**<add name="Cache-Control" value="no-cache, no-store" />**

**<add name="server" value="not available" />**

**</customHeaders>**

**</httpProtocol>**

**</system.webServer>**

**Cookie Same-Site Attribute Not Set**

**Go to Web.config file do the below changes.**

**<rewrite>**

**<outboundRules>**

**<rule name="AddSameSiteCookieFlag">**

**<match serverVariable="RESPONSE\_Set-Cookie" pattern="^(.\*)(CFID|CFTOKEN|JSESSIONID)(=.\*)$" />**

**<action type="Rewrite" value="{R:0};SameSite=lax" />**

**</rule>**

**</outboundRules>**

**</rewrite>**

**Missing Path Attribute In Session Cookie**

**<configuration>**

**<system.web>**

**<!-- Prevent access to cookies from other sub-domains -->**

**<httpCookies domain="host.\*.com" />**

**</system.web>**

**</configuration>**

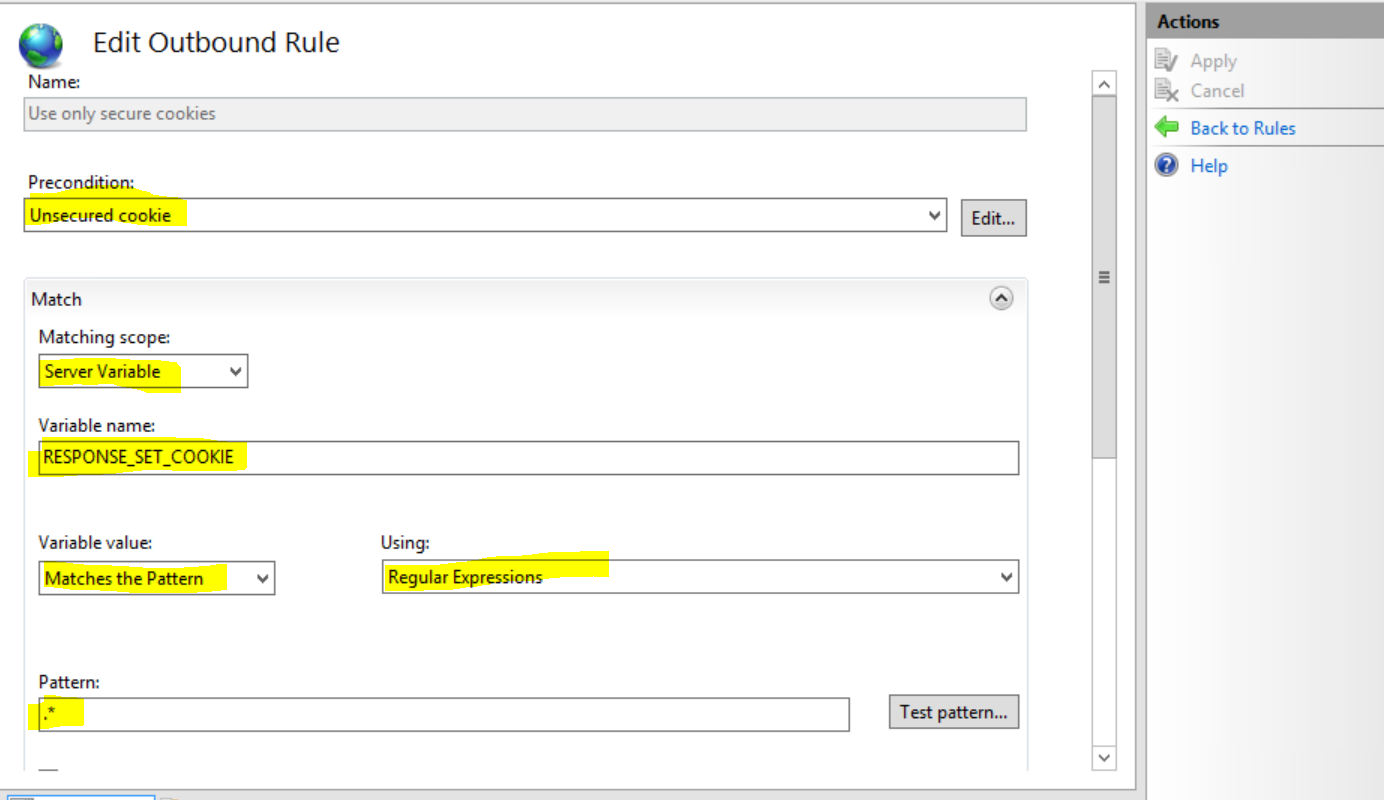
**<httpCookies httpOnlyCookies="true" requireSSL="true" />**

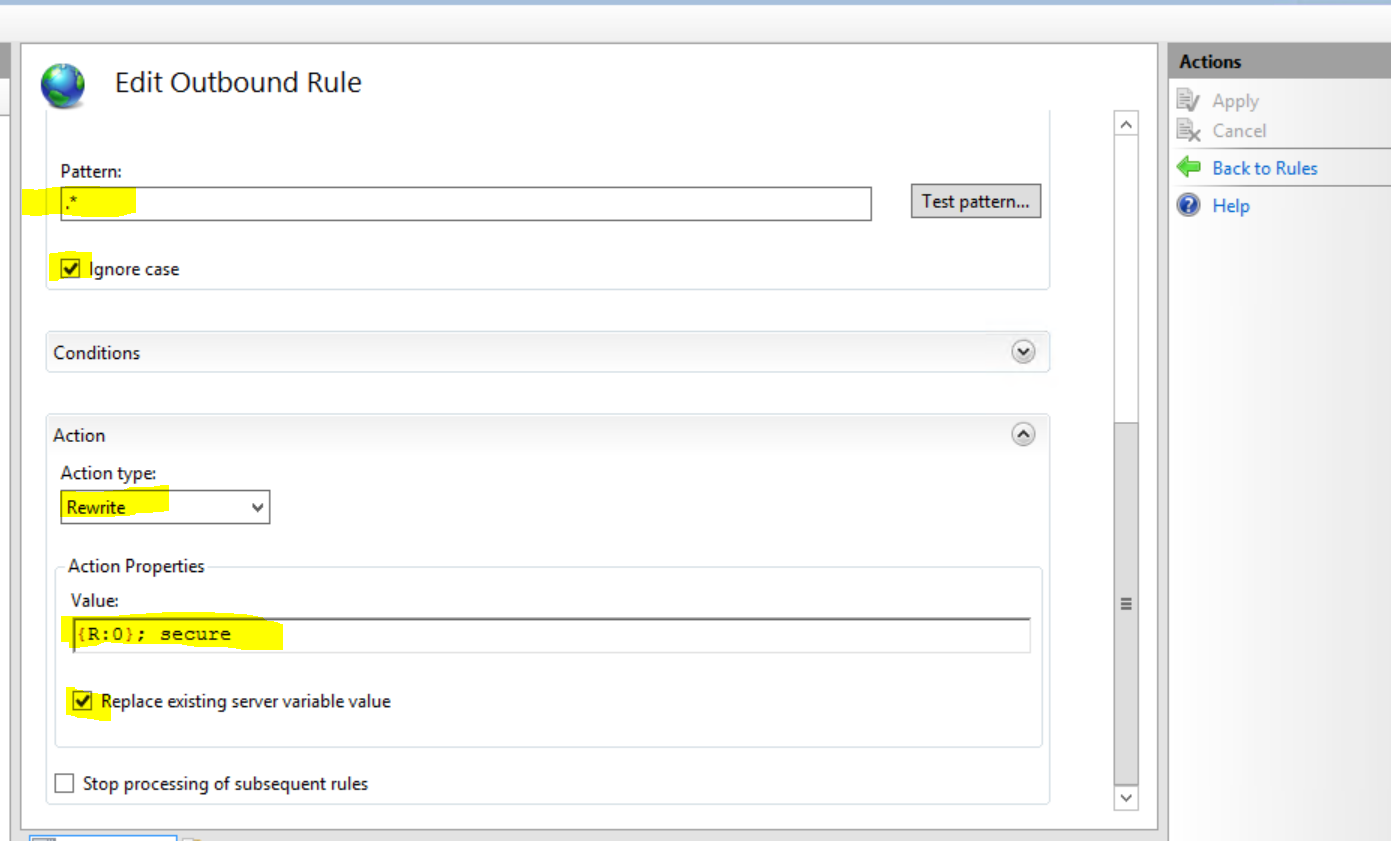
**<httpCookies domain="domain.com" httpOnlyCookies="true" requireSSL="true" />**

**<sessionState cookieName="Product"></sessionState>**

**Session cookie attributes are not set properly (8.5 IIS)**

In the URL Write module 🡺 add new rule and refer below screenshot.





**<rewrite>**

**<outboundRules>**

**<rule name="AddSameSiteCookie">**

**<match serverVariable="RESPONSE\_Set-Cookie" pattern=".\*" />**

**<action type="Rewrite" value="{R:0};SameSite=Strict" />**

**</rule>**

**</outboundRules>**

**</rewrite>**

**</system.webServer>**

**<system.web>**

**<authentication>**

**<forms cookieSameSite="Strict" requireSSL="false" />**

**</authentication>**

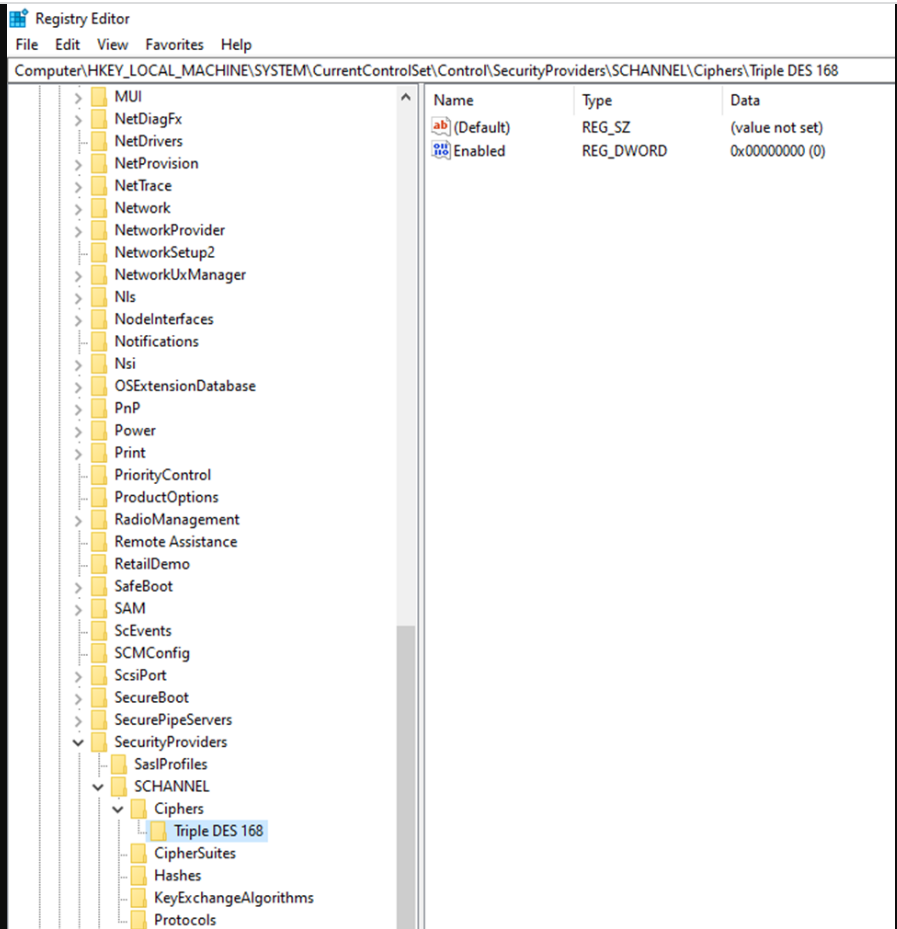
**SSL Medium Strength Cipher Suites Supported (SWEET32)**

Disable 3DES

To disable 3DES on your Windows server, set the following registry key:

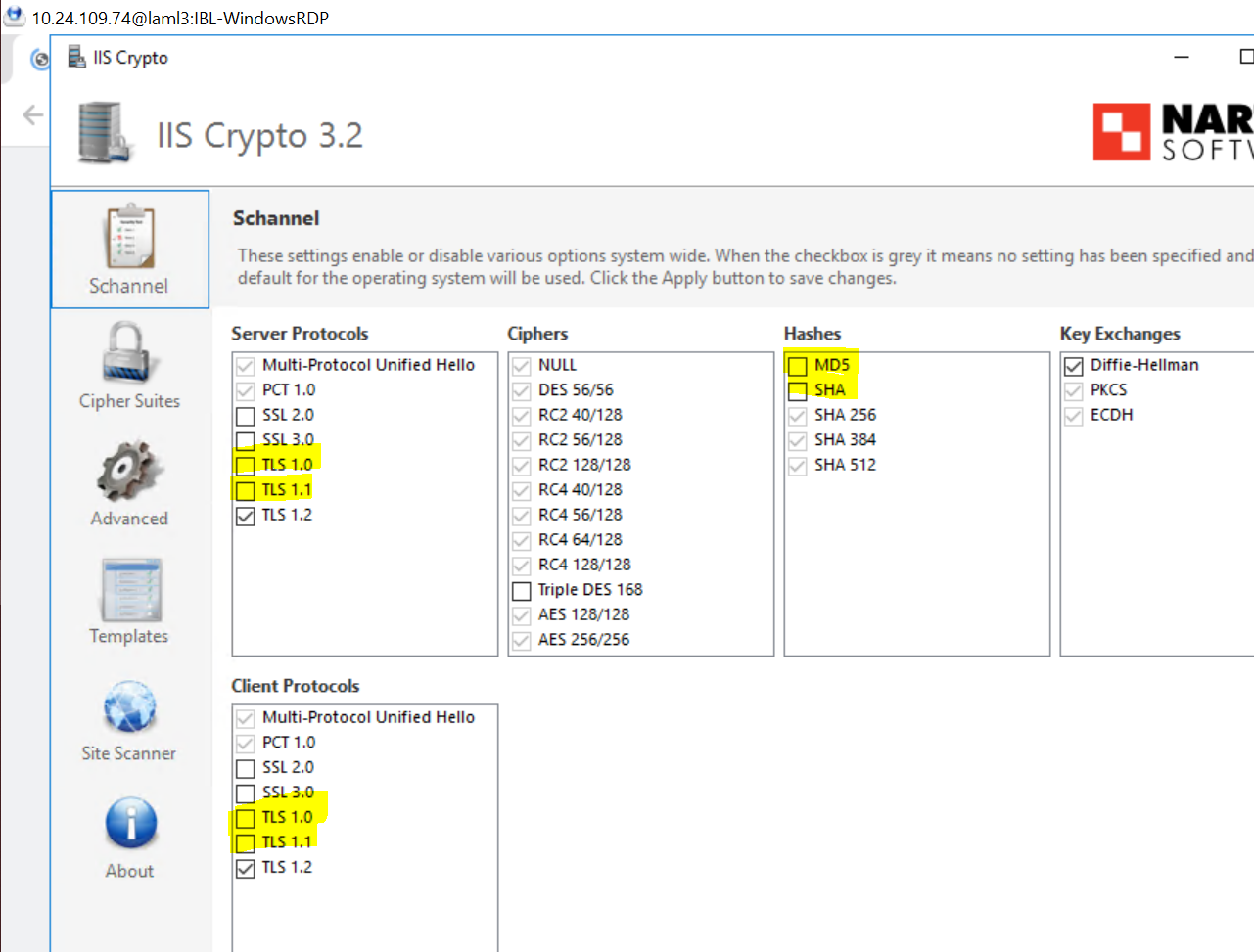
[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Ciphers\Triple DES 168]

“Enabled”=dword:00000000



SSL/TLS Certificate Supports Older version, CBC and Weak Cipher Algorithms

Reconfigure the affected application, to avoid the use of weak ciphers such as SSL, MD5, SHA1, RC4, 3DES, Weak algorithms like TLS1.0, TLS1.1. Use only TLS 1.2



SSL/TLS: Report Weak Cipher Suites (tcp/587)

Removed below mentioned Cipher form server.

TLS\_RSA\_WITH\_RC4\_128\_SHA

Diffie-Hellman Ephemeral Key Exchange DoS Vulnerability (SSL/TLS, D(HE)ater) (tcp/25)

Removed below mentioned Cipher form server

TLS\_DHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256

TLS\_DHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384

**The IIS default page is available**

**Open IIS Manager, go to Sites | Default Web Site🡺 double-click Default Document🡺click on Default.htm🡺 Click on Disable.**

