**IIS : security fix**

**Take IIS backup before IIS setting changes**

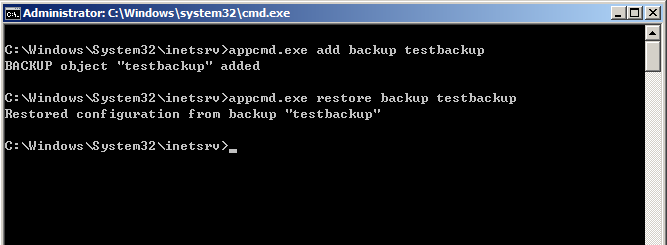
To back up and restore IIS 7 or IIS 8:

1. Open cmd.exe.
2. Navigate to %windir%\system32\inetsrv\.
3. Execute following command to back up configuration:

appcmd.exe add backup <backupname>.

1. Open the %windir%\system32\inetsrv\backup directory and copy the backup folder to a local folder.
2. Reinstall IIS.
3. Open the local folder and copy the backup folder to the %windir%\system32\inetsrv\backup directory.
4. Execute following command to back up configuration:

appcmd.exe restore backup <backupname>

[](https://esupport.trendmicro.com/media/13796015/cmd.png)

1. **X-XSS-Protection**

Implementing HTTP security headers are an important way to keep your site and your visitors safe from attacks and hackers. In a previous post, we dove into how the [X-Frame-Options](https://www.keycdn.com/blog/x-frame-options) header and frame-ancestors directive can help combat clickjacking.

**Enable on IIS**[**#**](https://www.keycdn.com/blog/x-xss-protection#enable-on-iis)

To enable on IIS simply add it to your site’s Web.config file.

<system.webServer>

...

<httpProtocol>

<customHeaders>

<add name="X-XSS-Protection" value="1; mode=block" />

</customHeaders>

</httpProtocol>

...

</system.webServer>

1. **Content-Security-Policy**

<system.webServer>

<httpProtocol>

**<customHeaders>**

**<add name="Content-Security-Policy" value="default-src 'self';" />**

**</customHeaders>**

</httpProtocol>

</system.webServer>

1. **Public-Key-Pins**

openssl x509 -in SDSSAR.Nestle.in.crt -pubkey -noout | openssl rsa -pubin -outform der | openssl dgst -sha256 -binary | openssl enc -base64

SDSSAR.Nestle.in

Fingerprint SHA256: 2a3485bb9b62aeafe09a547779dde4fae071b94479b39de942a7e0d13ed4f718

Pin SHA256: xBO4wasXO/WYJ7XeSZY0R8dAwuj1GqLtpzCwEq9Um7c=

RSA 2048 bits (e 65537) / SHA256withRSA

DigiCert SHA2 Secure Server CA

Fingerprint SHA256: 154c433c491929c5ef686e838e323664a00e6a0d822ccc958fb4dab03e49a08f

Pin SHA256: 5kJvNEMw0KjrCAu7eXY5HZdvyCS13BbA0VJG1RSP91w=

RSA 2048 bits (e 65537) / SHA256withRSA

In trust store DigiCert Global Root CA Self-signed

Fingerprint SHA256: 4348a0e9444c78cb265e058d5e8944b4d84f9662bd26db257f8934a443c70161

Pin SHA256: r/mIkG3eEpVdm+u/ko/cwxzOMo1bk4TyHIlByibiA5E=

RSA 2048 bits (e 65537) / SHA1withRSA

Weak or insecure signature, but no impact on root certificate

<httpProtocol>

<customHeaders>

<add name="Public-Key-Pins"

value="pin-sha256=”xBO4wasXO/WYJ7XeSZY0R8dAwuj1GqLtpzCwEq9Um7c=”;

pin-sha256=”5kJvNEMw0KjrCAu7eXY5HZdvyCS13BbA0VJG1RSP91w=”;

pin-sha256=”r/mIkG3eEpVdm+u/ko/cwxzOMo1bk4TyHIlByibiA5E=”;

max-age=5184000; includeSubDomains;" />

</customHeaders>

</httpProtocol>

Strict-Transport-Security

<configuration>

<system.webServer>

<rewrite>

<rules>

<rule name="Redirect HTTP to HTTPS" stopProcessing="true">

<match url="(.\*)" />

<conditions>

<add input="{HTTPS}" pattern="off" />

</conditions>

<action type="Redirect" url="https://{HTTP\_HOST}/{R:1}" redirectType="Permanent" />

</rule>

</rules>

<outboundRules>

<rule name="Add the STS header in HTTPS responses">

<match serverVariable="RESPONSE\_Strict\_Transport\_Security" pattern=".\*" />

<conditions>

<add input="{HTTPS}" pattern="on" />

</conditions>

<action type="Rewrite" value="max-age=31536000" />

</rule>

</outboundRules>

</rewrite>

</system.webServer>

</configuration>

HTTP response headers

## Remove Server response header with an outboundRule URL Rewrite rule

* 1. **Install URL Rewrite in IIS server**
  2. **Add the below line in web.config**

<rewrite>

<outboundRules rewriteBeforeCache="true">

<rule name="Remove Server header">

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

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

</rule>

</outboundRules>

</rewrite>

## Remove ASP.NET X-Powered-By header in IIS using web.config customHeaders

1. This response header can be removed with a customHeaders setting in web.config, placed in the <system.webServer> node:

<httpProtocol>

<customHeaders>

<remove name="X-Powered-By" />

</customHeaders>

</httpProtocol>

### X-AspNet-Version header

The X-AspNet-Version HTTP Header broadcasts to the world what version of [ASP.NET](https://www.saotn.org/code-base/) is being used. Add the following content inside the <system.web> node in your application’s web.config file:

<system.web>

<httpRuntime enableVersionHeader="false" />

</system.web>

<configuration>

<system.web>

<httpRuntime enableVersionHeader="false" />

</system.web>

<system.webServer>

<httpProtocol>

<customHeaders>

<remove name="X-Powered-By" />

</customHeaders>

</httpProtocol>

<rewrite>

<outboundRules rewriteBeforeCache="true">

<rule name="Remove Server header">

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

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

</rule>

</outboundRules>

</rewrite>

</system.webServer>

</configuration>