

General IIS 7.0 - .Net Questions

By [Walter Oliver](#)

December 22, 2007

If a module keeps crashing, does IIS 7.0 have ability to remove the "erring" module and recycle the app-pool without that module?

This must be done manually.

What is the impact of turning off kernel mode cache?

The impact will depend on the server load. If you are doing 10,000 requests/second, you will sorely miss kernel caching; however, if you are doing 100 requests/second you probably will not notice. It depends heavily on the content being served, IIS 5.0 did not have a kernel mode component and it worked well for most customers.

What support does Windows Server® 2008 have for Ruby on Rails?

There is basic ruby support. See following link for more information:

<http://mvolo.com/blogs/serverside/archive/2007/02/18/10-steps-to-get-Ruby-on-Rails-running-on-Windows-with-IIS-FastCGI.aspx>

Does the installation order of modules matter?

If you are installing via the Server Manager GUI ("Add Features"), order of installation does not matter. The Add Features wizard checks all dependencies and will alert you if you are missing any required modules. In addition, Add Features wizard knows the correct ordering of modules.

If you are installing from a command line or using an unattended installation, the order of modules doesn't matter (again, Setup knows the correct ordering of modules), but you are responsible for identifying all dependencies. If you fail to include a required dependency, unattended/ command-line setup will fail.

Note: As opposed to the order of modules during setup, the order in which modules are arranged in the common pipeline (i.e., the order in which modules subscribe to notifications) is important. For example, if two modules subscribe to the same notification, the one first on the list gets notified first (With one exception-- the default IIS 7.0 modules should not have issues with reordering). For the authentication modules, it is advisable to keep the existing ordering because this will determine with which authentication scheme IIS challenges first. We order it from most secure to less secure. IE uses the first authentication scheme it understands and if you put a less secure authentication scheme first, IE chooses it instead of the more secure one.

What is the memory footprint of an application pool? Does it load the CLR?

An Application Pool that only serves static files with all features installed will have a footprint of 3 MB private bytes, 5 MB page file. (This is larger than IIS 6.0). Windows Server 2008 handles multiple application pools better than WS03. When ASP.NET requests are made we pre-load a small amount of the CLR during startup (~100kb) . The preload is configurable by a property on the ApplicationPool. It is called managedRuntimeVersion. The rest of the CLR (~8mb) will be loaded on the first ASPX request.

Do customers need to have a 32-bit application pool and 64-bit application pool with the Access customers in 32-bit application pools?

Access only works in 32-Bit application pools. Loading the user profile (loadUserProfile property on the AppPool) is an issue when Classic ASP is used because Access is using the temp directory which doesn't allow access to the anonymous user when the user profile is loaded.

What are the limitations for Windows Server 2008 Web Edition?

Windows Server 2008 web edition is much improved, and we have focused hard on removing the artificial limits. The final licensing is not yet complete, but we are planning to remove all hardware restrictions, allow 4x processors and 32GB RAM (on x64). SQL is allowed, and SharePoint will be installable on the SKU.

What support will Windows Server 2008 have for Front Page Server Extensions?

FPSE is no longer a part of Windows Server. We are working with a third party to create a download package for FPSE to run on Windows Server 2008/IIS 7.0. It does not have any new features or enhancements, only fixes to make it compatible.

Does Windows Server 2008 support in-place upgrades?

We recommend that Windows Server 2008 be installed fresh and migrated to; or, just put new customers on new servers. We suggest that a well managed list of third party components and configurations is documented for each server, so that the current environment can be replicated on the new server. See the recommendations in the shared hosting paper for site and application pool configuration. A tool to assist in the migration will be released in the near future.

In-place upgrades are supported for the following scenarios:

- Windows Server 2003 can be upgraded to WS2K8 Beta3, WS2K8 RC0, WS2K8 RC1, and WS2K8 RTM
- Windows Server 2008 Beta3 can be upgraded to WS2K8 Beta3, and WS2K8 RC0
- Windows Server 2008 RC0 can be upgraded to WS2K8 RC0, WS2K8 RC1, and WS2K8 RTM
- Windows Server 2008 RC1 can be upgraded to WS2K8 RC1, and WS2K8 RTM

If the server is in Shared Configuration mode, it must be reverted to standalone configuration before the upgrade is run. To do so, disable shared config, copy down the applicationhost.config and encrypted keys to the local machine, run the upgrade on each server, then re-enable shared config.

Is it possible to specify a log file to be used during unattended setup? If so, is it possible to be granular on what is or is not logged?

Both the log files that setup is writing and the iis7.log are always on. It is not granular about what gets logged.

Is it possible to specify 3rd-party modules for use by pkgmgr during an unattended setup?

We do not provide any way to configure modules other than Windows modules during setup. There may be a way through generic unattend setup to run something after setup is done, and in that a user could do some coding.

What is the performance hit for failed request tracing? Is it possible to do failed request tracing for all web sites on a particular server?

Tracing ALL requests at <1000 requests/second should be <5% CPU. It is possible to configure a global tracing rule for all sites. Tracing can be enabled for all sites by changing the <siteDefaults> section.

Is it possible to limit the amount of memory an application pool will use?

No, but there is memory-based recycling, which will recycle AppPools that exceed configured memory limits.

Will the credentials encrypted with the machine key will be lost as a result of sysprep? Is there any workaround for this?

Encryptions made before sysprep are lost after sysprep. There is no workaround.

How does shared configuration handle multiple machines dealing with encrypted credentials?

You can export the machine keys and import them into all the servers so that decryption works. The UI for Server Beta 3 includes a feature called Shared Configuration which will allow you to do that. Click Export... and it will encrypt the machine keys, copy them along with applicationHost.config and administration.config to a path. After that, from all the other machines you can select "Import..." and it will import the machine keys and point the config to the shared configuration.

Can both Classic and Integrated Managed Pipeline Mode be enabled at the same time? If so, can it be configured such that some applications use one, and some use the other?

Different AppPools can have different values for this setting. Applications can be assigned to different AppPools

When there is no default document located within a folder being requested (HTTP Error 403.14), the error lists the server version information as IIS 7.0. Can that be masked to avoid unwanted information disclosure?

The default file for handling this error is contained in \inetpub\custerr\en-us. The footer of the error contains "Server Version Information: Internet Information Services 7.0.", which of course can be removed or edited directly from the .htm file.

Can a remote IIS 7.0 server be provisioned using the Managed API?

The managed API (Microsoft.Web.Administration) has access to all of the settings that the native API does and has DCOM remoting support by using ServerManager.OpenRemote static method. You can set any configuration settings but for Beta 3 there is no support for runtime information such as the State of an AppPool or the list of requests or workerprocess.

What is URL Authorization? Why would it be used?

In previous IIS versions you had to control access via file system ACLs. This is tedious and there is no web interface to do it. With URL authorization, you can control access to URLs using the IIS User Interface or using web.config directly. Additionally, you can use non-windows identities, e.g. Membership users and roles provided by forms authentication.

What changes (if any) to applicationhost.config and web.config trigger a restart of ALL application pools?

Any data in the applicationPool section relevant to that app-pool (so either in applicationPoolDefaults or specific to that app-pool) will cause WAS to recycle the app-pool. Worker process can ask WAS to recycle app-pools based on certain config changes, currently the only one we do it for is globalModules, but this is not a closed list (as modules can ask for recycle based on config change).

Is there a native component of load balancing or clustering within Windows Server 2008 and/or IIS 7.0?

NLB is part of Windows Server 2008. It is essentially the same as it was in Windows 2003. To install NLB, go to Server Manager > Features > "Add Features" and select "Network Load Balancing" from the list. To configure NLB, you need to open a command prompt and run nlbmgr. This is the UI that existed in Windows 2003.

What is meant by configurable CPU usage?

If the system-wide CPU exceeds a threshold dynamic, compression will stop occurring, freeing up the CPU it was using. If the system-wide CPU drops below a different limit, dynamic compression will resume, saving bandwidth.

Why would I not enable dynamic compression all the time?

Different users have different opinions on optimal CPU utilization. Some think a 20% average is perfect, others think 75%. If you want X and you're consistently above X, you might as well completely remove dynamic compression, as it is never going to be used.

How does dynamic compressions factor in consistent CPU spiking vs. constant CPU usage?

It factors in average usage over the 30 second window since the last sample - so, even with irregular spikes, you will have dynamic compression on for at most 30 seconds after the spike - and if your spikes are instantaneous (and so do not affect average CPU usage over 30 seconds much) - they will not affect dynamic compression.

In a shared hosting environment, what should the default LoadUserProfile setting be?

Setting loadUserProfile=false in applicationPoolDefaults is a good idea for Shared Hosting scenarios. The startup time of an AppPool will be much faster and you avoid any temporary directory permission issues.

What is the cause of the "Http 500.19 - Internal Server Error"?

The 500.19 error is caused by the IIS 7.0 feature delegation mode. When a feature is delegated to site owners, and the site owners modify the feature, then their changes are persisted in web.config. If the server administration revokes the delegated management on that feature, then the web site owner has the responsibility of cleaning up the feature details from web.config. Otherwise, all sites that had modified that delegated feature will immediately give an "Http Error: 500.19 - Internal Server Error" message. In order to avoid this issue, we recommend that hosters do not revoke delegated features once they are published to end customers.

How does IIS 7.0 handle web.config updates?

If the hosted site does not have a web.config, IIS 7.0 will create one. If the site has a web.config, IIS

7.0 modifies it. If the web.config is modified, then the site owners have the responsibility of merging the changes and ensuring that the changes are manually merged and maintained.

At IIS 7.0 install time, we did not install the management service. What are the impacts of installing this service now that we are using shared configuration?

Installing Management Service does not modify AppHost.config or Administration.config at all. The only changes you should see are the new binaries (wmsvc.exe). A self-sign certificate will be created and a few registry keys will be added. This means that in theory nothing should break.

Does installing the management service make any changes to the shared configuration ACLing to make it work with the administration service?

For the most part, it works out of the box since we use the redirection.config settings for reading apphost/admon.config. However, for a detailed answer, it really depends on what scenarios you will be using:

- Shared Config using Local Content: Regular Windows or IIS users connecting to modify their local content and their web.config's - there is nothing to do, everything should work out of the box.
- Shared Config using Remote Content using Windows Users: It should work, provided the Windows accounts have access to their content.
- Shared Config using Remote Content using IIS Users: For this scenario you must change the Identity of the service (WMSVC) to an account that has access to the remote content, since we use the process identity for accessing content. Note that apphost.config/admon.config will work, since we use redirection.config
- Windows Administrator managing a Server Connection: It should work, provided your Windows administrator has write access to the shared config.

How can a web site be created from the command line?

See the following link for information about creating sites in IIS 7.0.

<http://technet2.microsoft.com/windowsserver2008/en/library/f6c26eb7-ad7e-4fe2-9239-9f5aa4ff44ce1033.mspx?mfr=true>

Is CLR loaded automatically for each w3wp/apppool?

An Application Pool that only serves static files with all features installed occupies 3 MB private bytes, 5 MB page file. When ASP.NET requests are made we pre-load a small amount of the CLR during startup (~100kb) . The preload is configurable by a property on the ApplicationPool. It is called managedRuntimeVersion. The rest of the CLR (~8mb) will be loaded on the first ASPX request.

When IIS 7.0 provisions a new web site, it creates folders like W3SVC1, FTPSVC2, etc and assigns permissions: Administrators - Full Control, SYSTEM - Full Control. As a result, those folders (and log files inside) are unavailable to the site user for download. Is it possible to override this IIS 7.0 behavior and force it to create log directories with permissions inherited from parent directory?

Http.sys creates these folders automatically if they do not exist. If you override the permissions with something else, it should preserve the new permissions.

Is IIS 7.0 compatible with ColdFusion 8?

We have tested it internally and it seems to work well if ISAPI and Metabase are installed. There are some blogs online as well that give instructions on how to make it work. For more information see the following link:

<http://blogs.iis.net/bills/archive/2007/03/06/coldfusion-on-iis7.aspx>.

After setting up IIS 7.0 with WSS3 and using central administration to create a site collection, why does the following error occur: "The page cannot be displayed because your server's current configuration does not support it. To perform this task, use the command line operations in Stsadm.exe."?

The server is setup in AD Account Creation mode. AD Creation Mode is a deprecated feature that is still supported, but will be removed in V4. Instead of allowing WSS to automatically create users in AD, the recommendation is to do user provisioning outside of WSS.

Where can I find more information about the Shared Centralized Global Configuration Feature?

More information on Centralized Global Configuration can be found at: [The Configuration System in IIS 7.0](#)

Are there FrontPage server extensions available for Windows Server 2008 64 bit? I only find a download for I386.

There are currently no FPSE for x64. Our shared hosting recommended architecture is 64bit OS with 32bit AppPools. Unfortunately there is currently a bug that prevents FPSE from installing on 64bit for this scenario.

Windows Server 2008 was installed without a Product Key and is now asking for an activation code. The Product Key is not available right now, how can I re-activate Windows Server 2008?

You may re-arm your system three times by completing the following steps:

In order to run `slmgr /rearm`, open `regedit.exe` and navigate to `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SL`. Verify that value "skiprearm" is set to '0'. If this value is non-zero, the re-arm function will not reset the system activation timers.

After verifying that the "skiprearm" registry value equals zero (0), run `slmgr /rearm` from an elevated command prompt. Wait for the notice that the process has completed. This can take a minute or two. Once complete, follow the prompt to shutdown the computer. Upon restart, the computer will be running in OOB Grace and will have another 30 days in which to activate. No other changes will be made to the system by this process.

What is the recommended method to deploy x509 certificates on multiple web servers?

IIS.CertObj COM-object is still there in IIS 7.0 and we think it is still the best option for deploying certificates on multiple web servers. This component behavior remains the same, so all old script should work (if ABOMapper is enabled).

Note: In LH RC0 there will be a new feature of this object that allows specifying secure bindings as an instance name:

```
set iiscertobj = CreateObject("IIS.CertObj")
```

```
iiscertobj.InstanceName = "0.0.0.0:443"
```

iiscertobj.Import pfxfile, pfxfilepassword, true, true

And such a script won't be depending on ABOMapper.

.NET Questions

By [Walter Oliver](#)

December 22, 2007

Can ASP.NET websites built with Visual Studio 2008 Beta 2 be used in partial trust configurations?

No. When a user uses Visual Studio 2008 Beta 2 to build a new ASP.NET 3.5 website, or migrates an existing ASP.NET website to .NET Framework 3.5, the application cannot be deployed and run on a server configured for medium trust or partial trust, as is recommended for ASP.NET Web hosting. The issue only affects sites that have been built with Visual Studio 2008 Beta 2 or Visual Web Developer 2008 Express Beta 2. It does not affect applications built for earlier versions of the .NET Framework, and has no impact on existing applications on a Web server, even after installing .NET Framework 3.5 on the computer.

A workaround is available for users to modify their websites so they can run correctly in partial trust. We will be publishing this workaround to users who download the affected products. If you are providing .NET Framework 3.5 hosting, you can also provide this information to users. This workaround requires no other action for hosters.

When a user builds a new ASP.NET website with Visual Studio 2008 Beta 2 or Visual Web Developer 2008 Express, these products insert a new entry in the applications web.config configuration file. This is done when the user creates a new project, or migrates an existing website to use .NET Framework 3.5. The configuration entry looks as follows:

```
<system.codedom> <compilers> <compiler language="c#;cs;csharp"
extension=".cs" compilerOptions="/warnaserror-" warningLevel="4"
type="Microsoft.CSharp.CSharpCodeProvider, System,
Version=2.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089"> <providerOption
name="CompilerVersion" value="v3.5"/> </compiler> <compiler
language="vb;vbs;visualbasic;vbscript" extension=".vb"
compilerOptions="/optioninfer+"
type="Microsoft.VisualBasic.VBCodeProvider, System,
Version=2.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089"> <providerOption
name="CompilerVersion" value="v3.5"/> </compiler> </compilers>
</system.codedom>
```

When this website is run in medium trust or any partial trust setting, ASP.NET will raise an error noting that "the current trust level does not allow use of the 'compilerOptions' attribute". The compilerOptions setting specified in the above configuration is not allowed in medium trust or a partial trust configuration.

Note: Depending on the security settings of your server, this error message may or may not be visible to the user.

As a workaround, when deploying a website built with Visual Studio 2008 Beta 2 or Visual Web

Developer 2008 Express to a partial trust configuration, users can simply remove the compilerOptions setting, as well as the warningLevel setting, from the web.config configuration file. The workaround is described in the attached document, which you can make available to users.

Using this workaround may also have other impact on the website; these issues are described in detail in the attached document.

Can LINQ features in .NET Framework 3.5 Beta 2 be used by default in partial trust configurations?

No. .NET Framework 3.5 Beta 2 includes a feature called Language Integrated Query (LINQ), which enables rich data capabilities in websites, and is a key new feature of .NET Framework 3.5. However, by default, LINQ does not function properly in medium trust and partial trust configurations, as is recommended for ASP.NET Web hosting. Enabling LINQ in these configurations requires the hoster to make a change to the trust policy settings on the Server.

In medium trust or partial trust configurations, the code permissions granted to an ASP.NET website are determined by a Code Access Security (CAS) policy file on the Web server. When .NET Framework 3.5 is installed on a Web server, websites continue to use the same CAS policy file as .NET Framework 2.0.

The LINQ feature set in .NET Framework 3.5 requires the CAS policy file to grant a new permission, called RestrictedMemberAccess, which is not granted by default on ASP.NET 2.0. To enable LINQ to work in medium or partial trust, you need to modify the CAS policy file to grant this additional policy.

Note: Making this change will also grant this policy to ASP.NET 2.0 websites running on the same server. We have determined this to be an acceptable change for hosted sites that run under medium trust. This change will have no impact on existing ASP.NET 2.0 websites that can run under medium or partial trust.

As a workaround, you can enable LINQ for medium trust by following these steps on the server:

1. Open a command prompt, and go to the directory that contains your ASP.NET 2.0 trust policy files. This is found under the Windows directory, at

```
%windir%\Microsoft.NET\Framework\v2.0.50727\config
```

2. Determine which CAS policy file to modify. If you are using medium trust, this file will be web_mediumtrust.config.

3. Make a backup of the existing file.

4. Examine the <SecurityClasses> section of your CAS policy file. If the section does not contain an entry named ReflectionPermission, add a new entry as follows:

```
<SecurityClass Name="ReflectionPermission"
Description="System.Security.Permissions.ReflectionPermission, mscorlib,
Version=2.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089"/>
```

If you are using the default medium trust settings, you must add this entry. Depending upon your trust setting, your CAS policy file may already contain this entry.

5. Examine the <NamedPermissionSets> section of your CAS policy file. If the section does not contain an entry named ReflectionPermission, add a new entry as follows:

```
<IPermission class="ReflectionPermission" version="1"
Flags="RestrictedMemberAccess" />
```

If you are using the default medium trust settings, you will need to add this entry.

If you are using another trust configuration, and the section already contains an entry named ReflectionPermission, you can modify the Flags setting to add RestrictedMemberAccess permission. Flags should be separated by a comma. For example, if you are using the default high trust settings, you can modify this section as follows:

```
<IPermission class="ReflectionPermission" version="1"
Flags="ReflectionEmit, RestrictedMemberAccess" />
```

6. Save the file, and restart the Web server.

To verify whether the changes have been made successfully, create a new ASP.NET website containing the files in the enclosed ZIP, and run it.

Known Issue: Installation Order for .NET 3.0

There is an issue with how .NET 3.0 is installed on IIS 7.0 / Windows Server® 2008 when shared hosting is enabled. The installer ignores the fact that shared hosting may be enabled on the machine and always updates the local applicationhost.config of the machine and not shared UNC one. This issue is mitigated if .NET 3.0 is installed before shared hosting is enabled and the common applicationhost.config deployed. The issue is just with enabling .NET 3.0 the first time (registering the modules/handlers), all subsequent operations (changing bindings, app pools, etc) are reflected correctly against the shared config file thereafter.

Is .NET 1.1 included in 3.5?

Yes. Any prior version of .NET (from 1.0 to 3.0) is included as part of .NET 3.5. .NET 3.5 is a comprehensive package.

Does the .NET 3.5 Framework include Entity Framework Beta 2?

No. It doesn't include Entity Framework beta2. You need to download here:

<http://www.microsoft.com/downloads/details.aspx?FamilyId=F1ADC5D1-A42E-40A6-A68C-A42EE11186F7&displaylang=en>

Here is the release info:

<http://blogs.msdn.com/adonet/archive/2007/08/27/entity-framework-beta-2-the-1st-entity-framework-tools-ctp-released.aspx>

Entity Framework will not be available until first half of 2008:

<http://blogs.msdn.com/adonet/archive/2007/04/28/ado-net-entity-framework-update.aspx>

Top 10 Changes in IIS 7.0

By [Walter Oliver](#)

December 22, 2007

1. Simple, configurable command line setup

Install only the IIS components needed to run your site

Example:

```
start /w pkgmgr /l:log.etw /iu:IIS-WebServerRole;IIS-WebServer;IIS-CommonHttpFeatures;IIS-StaticContent;IIS-DefaultDocument;IIS-DirectoryBrowsing;IIS-HttpErrors;IIS-HttpRedirect;IIS-ApplicationDevelopment;IIS-ASPNET;IIS-NetFxExtensibility;IIS-ASP;IIS-ISAPIExtensions;IIS-ISAPIFilter;IIS-ServerSideIncludes;IIS-HealthAndDiagnostics;IIS-HttpLogging;IIS-LoggingLibraries;IIS-RequestMonitor;IIS-HttpTracing;IIS-Security;IIS-ClientCertificateMappingAuthentication;IIS-IISCertificateMappingAuthentication;IIS-RequestFiltering;IIS-IPSecurity;IIS-Performance;IIS-HttpCompressionStatic;IIS-HttpCompressionDynamic;IIS-WebServerManagementTools;IIS-ManagementConsole;IIS-ManagementScriptingTools;IIS-ManagementService;IIS-IIS6ManagementCompatibility;IIS-Metabase;IIS-WMICompatibility;IIS-LegacyScripts;IIS-LegacySnapIn;WAS-WindowsActivationService;WAS-ProcessModel;WAS-NetFxEnvironment;WAS-ConfigurationAPI
```

2. Great compatibility story

- Most (99%+) ASP and ASP.NET applications worked.
 - One application encountered breaking change
 - Handful of applications required config migration to run in Integrated

(We have about 260 applications running on www.microsoft.com as defined by IIS. There are thousands of pages of code that could have broken but did not.)

- [Integrated Pipeline](#) is the new unified request processing pipeline. Benefits include:
 - Allowing services provided by both native and managed modules to apply to all requests, regardless of handler. For example, managed Forms Authentication can be used for all content, including ASP pages, CGIs, and static files.
 - Empowering ASP.NET components to provide functionality that was previously unavailable to them due to their placement in the server pipeline. For example, a managed module providing request rewriting functionality can rewrite the request prior to any server processing, including authentication, takes place.
 - A single place to implement, configure, monitor and support server features. For example, single module and handler mapping configuration, single custom errors configuration, single URL authorization configuration.
- Classic ASP mode allows for easy app migration
 - ASP.NET Setup provides a "Classic .NET AppPool"
 - For more information, see the article [ASP.Net Integration With IIS 7.0](#)

- Use [AppCmd](#) to migrate apps to Integrated mode
 - %windir%\system32\inetsrv\APPCMD.EXE migrate config <Application Path>
 - For more information about AppCmd.exe, see [Getting Started With AppCmd.exe](#)
- [IIS 6.0 Metabase compatibility](#) layer
 - Allows you the run old ADSI scripts
 - IIS 6.0 Metabase Compatibility module must be installed

3. No more metabase!

- Clean clear-text schema
- IIS settings stored in XML configuration file (applicationHost.config)
 - Metabase exists for SMTP/NNTP/FTP only
- [Site-wide changes made easily](#)
 - Update central applicationHost.config and copy to all web servers
 - Replaces our bulky ADSI based script solution for metabase changes
- Microsoft.com considerations
 - Be careful copying to production servers under load. (When you push out a new applicationHost.config, your app pools and applications will be recycled. If you are heavily dependent on caching, then you could cause problems in your environment as the CLR reloads.)

4. Centralized configuration

- applicationHost.config stored on UNC share
- Allows us to copy to two (maybe four) servers rather than 80

Potential issue - managing password changes for account used to connect to config store. (This is because currently you cannot use the UNC share that is running under the Network service, which we use heavily. It currently requires a domain account, which our security policy mandates a periodic password change.)

5. Delegated configuration

- Admin can now delegate IIS settings to application owner
- Settings defined in web.config file in application directory
- Example of setting to delegate include:
 - System.webServer section of applicationHost.config
 - Caching, defaultDocument, httpErrors, security

For more information, see [Delegated Configuration](#) .

6. AppCmd and other new management options

- Managing via the UI
 - New modular, task-based look and feel
 - Moving away from the right-click/properties paradigm
- Managing via the Command Line
 - AppCmd
- Command line utility which replaces adsutil.vbs, iisapp.vbs, and others
- Allows command line management of sites, applications, vdirs, apppools, modules, tracing, and more Powershell
- IIS community creating IIS-specific Powershell cmdlets
- MSCOM Considerations
 - AppCmd limitations - no remote

- No IIS provider for Powershell

7. Failed request tracing

- Buffers the trace events for requests and flushes them to disk if they meet your failure criteria
- Captures trace data while you are sleeping
- Very little perf impact when targeting failing requests
- Quick test: Enabling tracing for all file extensions and errors results in approx 5% fewer requests/second at full stress load (do not do this in production).
- View Currently Executing Requests via AppCmd
 - `appcmd list requests` (for all request)
 - `appcmd list requests /apppool.name:DefaultAppPool`
- New Task Scheduler
 - Trigger tasks on events

For more information, see [Failed Request Tracing](#).

8. Request filtering

- No more URLScan
- `</requestFiltering>` settings in `applicationHost.config`
- Issue for Microsoft.com: If filename includes "+" then `allowDoubleEscaping` must be set to "true"
 - `<requestFiltering allowDoubleEscaping="true">`
- Allow or disallow specific file extensions and verbs
 - `<add fileExtension=".exe" allowed="false" />`
- DenyURLSequences
 - `<add sequence="." />`
 - `<add sequence="/" />`
- RequestLimits
 - `maxAllowedContentLength="1000000"`
 - `maxUrl="260"`
 - `maxQueryString="2048"`

9. UNC content

- Simplified content synchronization
- Reduced H/W footprint (potentially less cost)
 - Common industry pain point

10. Output caching of dynamic content

- Fewer off-box calls to back end dependencies
- Significant performance gains
- Simple WCAT (Web Capacity Analysis Tool) Stress Test against www.microsoft.com/en/us/default.aspx

Not appropriate for all applications (e.g., not effective for those with very personalized output)