Internet Information Services (IIS)

Server OS

Contents

- A brief history of IIS
- Sites, Applications and Virtual Directories
- Bindings
- Application Pools
- Authentication
- Installation

A brief history of IIS

- IIS 1.0 → Windows NT 3.51 (30/05/1995)
- IIS 2.0 → Windows NT 4.0 (24/08/1996)
- IIS 3.0 → Windows NT 4.0 SP2 (14/12/1996)
- IIS 4.0 → Windows NT 4.0 Options Pack (around 1998)
- IIS 5.0 → Windows 2000 (17/02/2000)
- IIS 5.1 → Windows XP Professional (25/10/2001)

A brief history of IIS

- IIS 6.0 → Windows Server 2003 (24/04/2003)
  - IPv6
  - Worker Process Model
  - HTTP protocol listener
  - Authentication methods
- IIS 7.0 → Windows Server 2008 (04/02/2008)
  - Complete redesign and rewrite of IIS
  - Modules
  - Enhanced security and performance

A brief history of IIS

- IIS 7.5 → Windows Server 2008 R2 (22/10/2009)
  - Improved modules (ex. FTP)
  - PowerShell CLI
  - TLS 1.1 and 1.2
  - Best Practice Analyzer tool
- IIS 8.0 → Windows Server 2012 (04/09/2012)
  - Centralized SSL certificates
  - SSL binding to hostname
  - Application Initialization (+splash page)
  - ASP.NET 4.5

A brief history of IIS

- IIS 8.5 → Windows Server 2012 R2 (18/10/2013)
  - Automatic Certificate Rebind
  - Idle worker-Process page-out
  - Dynamic Site Activation

- IIS 10.0 → Windows Server 2016 (12/10/2016)
  - HTTP/2
  - Windows containers on Nano Server
  - New PowerShell cmdlets

Sites, Applications and Virtual Directories

- Sites
  - A site is a container for applications and virtual directories
  - You can access it through one or more unique bindings
    - Hostname, ip address and port
  - Configuration settings:
    - Limits
      - amount of bandwidth
      - the number of connections
      - amount of time allowed for connections to a site
    - Logging
      - handling and storage of log files
      - Failed request trace logs

https://docs.microsoft.com/en-us/iis/get-started/planning-your-iis-architecture/understanding-sites-applications-and-virtual-directories-on-iis

Sites, Applications and Virtual Directories

- Applications
  - Group of files that delivers content or provides services
  - The application's path becomes part of the site's URL
  - Each site must have an application
    - Root application or default application
  - An application belongs also to an application pool
  - Example: Online commerce website that has several applications
    - Shopping cart application
    - Login application
    - Search application

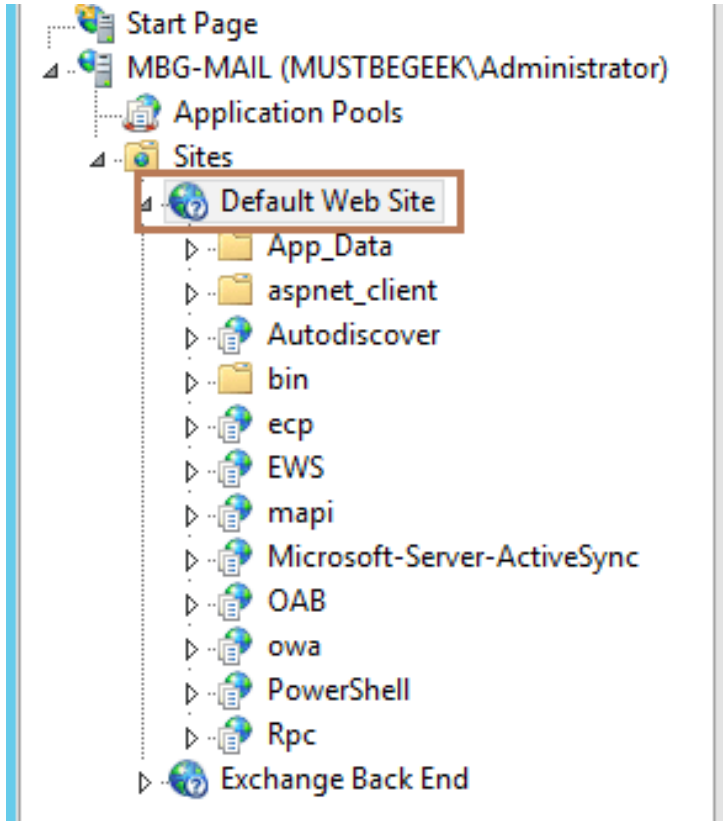
https://docs.microsoft.com/en-us/iis/get-started/planning-your-iis-architecture/understanding-sites-applications-and-virtual-directories-on-iis

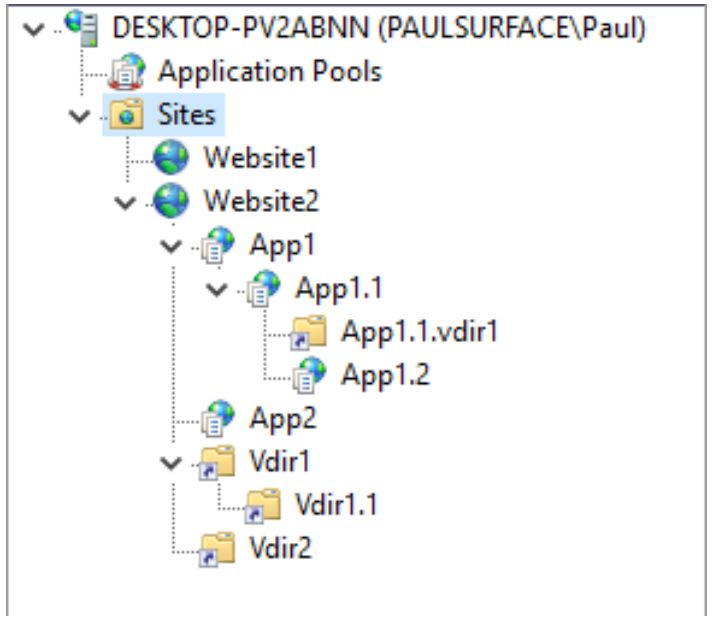
Sites, Applications and Virtual Directories

- Virtual Directories
  - Directory name (or path)
    - Mapping to a physical directory on a local or remote server
    - Becomes part of the application's URL
  - Each application must have a virtual directory
    - Root virtual directory
    - Maps the application to the physical directory that contains the application's content
  - An application can have more than one virtual directory
  - Example:
    - Include images from another location in the file system
    - No need to move the images

https://docs.microsoft.com/en-us/iis/get-started/planning-your-iis-architecture/understanding-sites-applications-and-virtual-directories-on-iis

Sites, Applications and Virtual Directories





erasmus HOGESCHOOL BRUSSEL

Sites, Applications and Virtual Directories

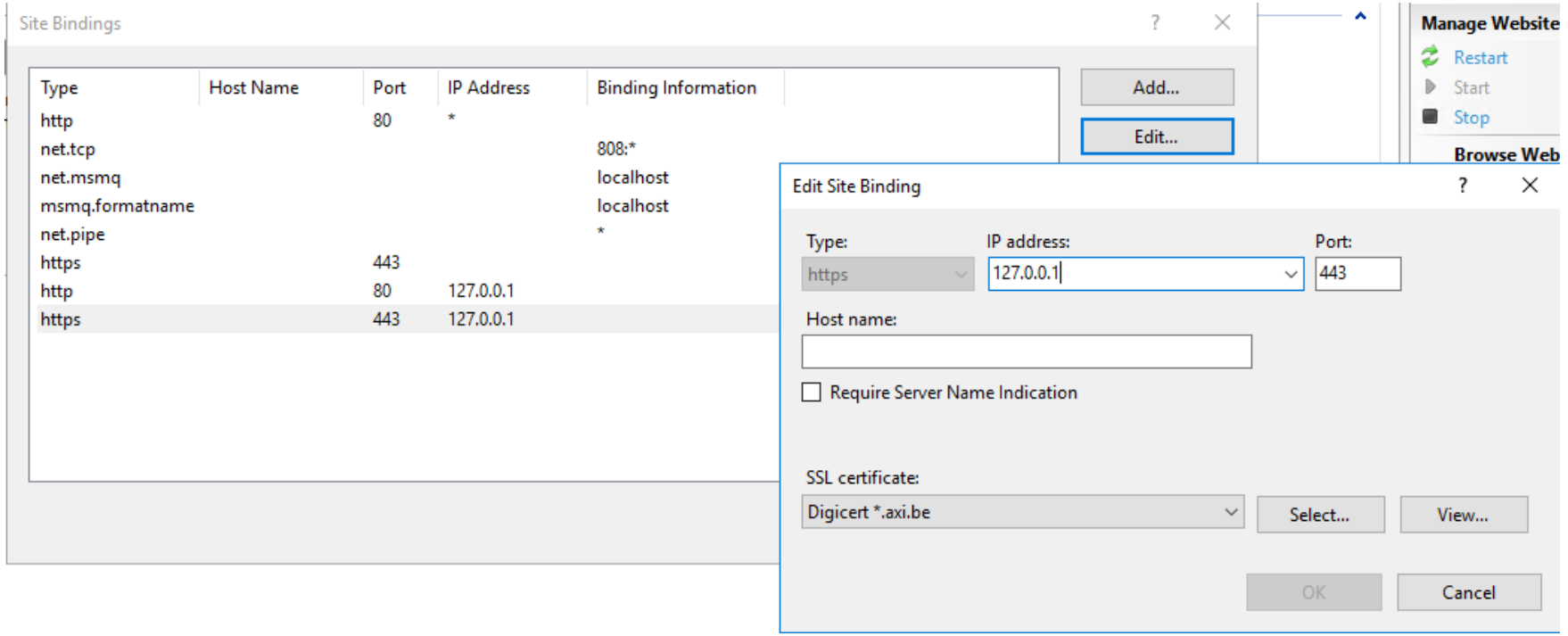
- ApplicationHost.config file
  - Located at %windir%\system32\inetsrv\config\

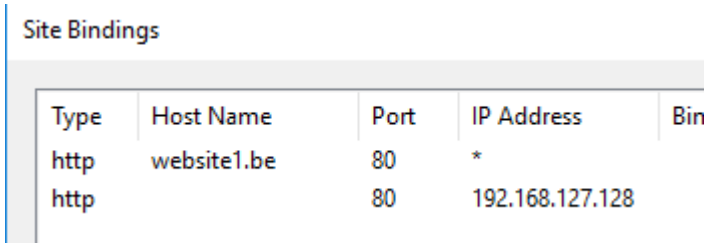
```
<sites>
    <site name="Default Web Site" id="1">
        <application path="/">
            <virtualDirectory path="/" physicalPath="%SystemDrive%\inetpub\wwwroot" />
        </application>
        <bindings>
            <binding protocol="http" bindingInformation="*:80:" />
        </bindings>
 </site>
```

Bindings

- Binding protocol
  - Defines the protocol over which communication between the server and client occurs
  - HTTP(S), FTP, …
  - (Listen in Apache/Nginx)
- Binding information
  - Defines the information that is used to access the site
  - IP address, port, host name
- A site may contain more than one binding

Bindings





erasmus HOGESCHOOL BRUSSEL

Application Pools

- Containing a single or multiple applications
- DefaultAppPool
  - Automatically created when installing IIS
  - Every application will run here if no other AppPool is created
- One or more Worker Processes
  - Windows process
  - w3wp.exe
  - Handles the web requests for the application pool
  - At least one per AppPool
  - More are allowed

https://docs.microsoft.com/en-us/iis/configuration/system.applicationhost/applicationpools/
https://stackify.com/w3wp-exe-iis-worker-process/

Application Pools

- Isolation of web applications
  - See Docker → Microservices
  - If maintenance is needed, only one part of the website is down
- More security
  - Applications do not communicate
  - Different Identity accounts
    - Account with the name of the application pool
    - Run the application pool's worker processes under this account
- Each application pool has its own settings

https://docs.microsoft.com/en-us/iis/configuration/system.applicationhost/applicationpools/

Application Pools

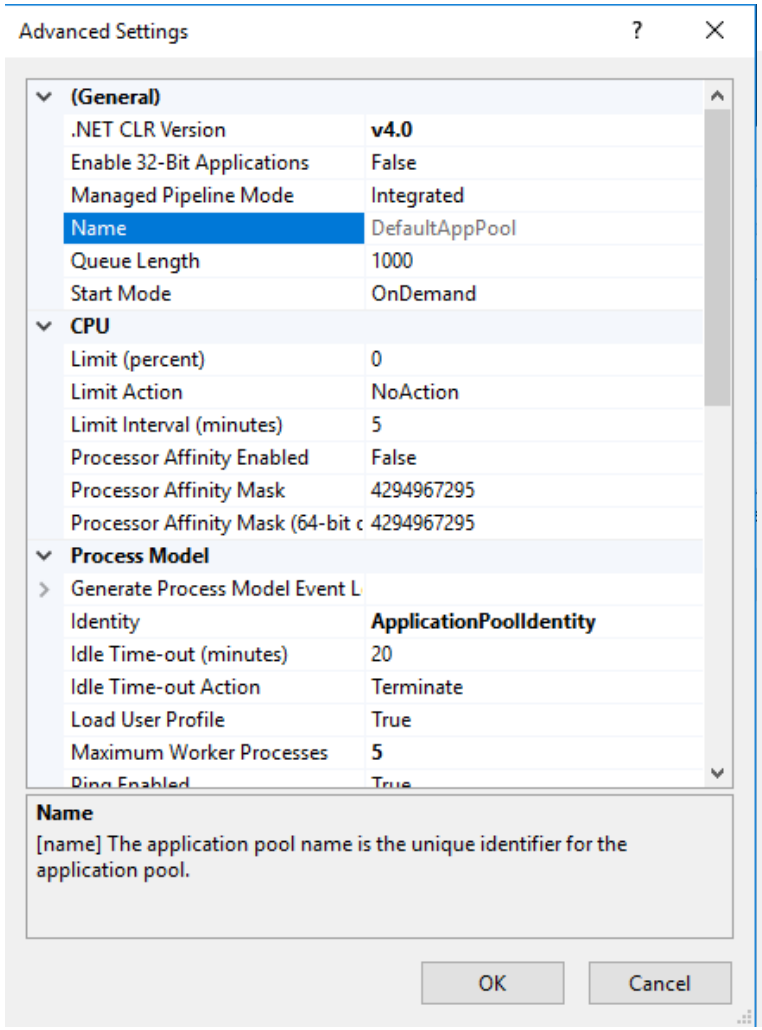
- Recycling Settings
  - How often the App will be recycled such as by time intervals, memory usage, etc.
  - Worker process is terminated and a new one starts
  - Avoid unstable states
    - application crashes, hangs, or memory leaks
  - Default: overlapped recycle method
    - Start new process to handle new requests
    - Keep the old process up to handle existing requests
      - Until finished
      - Until set timeout

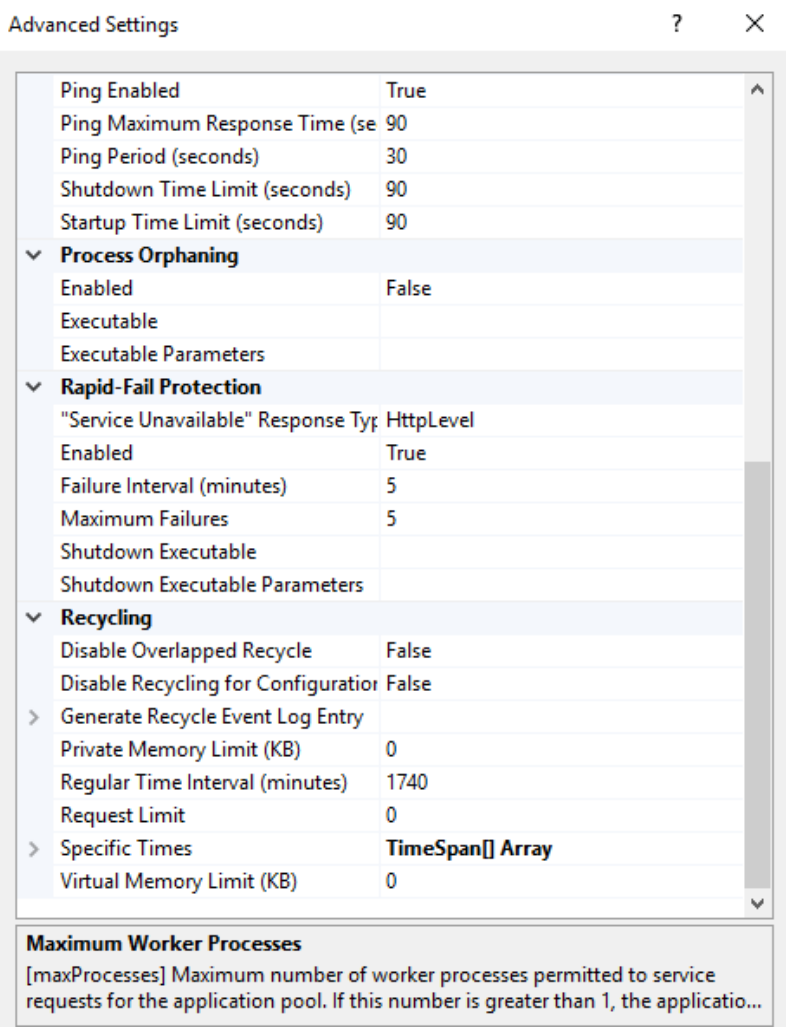
Application Pools

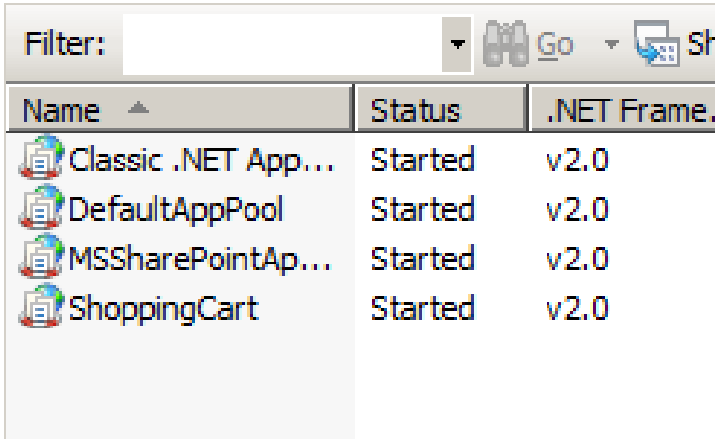
- Pipeline type
  - .NET integration modes
  - How IIS processes an incoming request
  - Integrated pipeline:
    - IIS 7 and later
    - ASP.NET 2.0
    - Static content, PHP and other content types
  - Classic pipeline:
    - Not as efficient as Integrated
    - IIS 6.0 processing pipeline
    - Run ASP.NET version 1.1 applications on an IIS 7 and later without modifying the application

https://docs.microsoft.com/en-us/iis/configuration/system.applicationhost/applicationpools/

Application Pools







erasmus HOGESCHOOL BRUSSEL

Authentication

- The web **browser** makes a request, such as HTTP-GET
- The web **server** performs an authentication check
  - If this is not successful because authentication is required, the server responds with an error
    - 401: You are not authorized to view this page
    - 403: You do not have permission to view this directory or page using the credentials you supplied
- The web **browser** constructs a new request that contains authentication information
- The web **server** performs an authentication check
  - Successful: the server sends the data to the browser

https://support.microsoft.com/en-us/help/324276/how-to-configure-internet-information-services-web-authentication-in-w

Authentication

- Anonymous Authentication
  - Default: IUSR_**ComputerName** account
  - Log on locally
  - Use any valid windows account
  - You can set up different anonymous accounts for different Web sites, virtual directories or physical directories, and files.
- Basic Authentication
  - User must enter credentials
  - Access is based on the user ID
  - Clear text!

Authentication

- Windows Integrated Authentication
  - More secure than basic
  - Intranet use
    - Client computers and Web servers are in the same domain
    - Connection to AD
  - Use the current user's credentials from a domain logon
    - Failed: prompted to enter a username and password
  - The user's password is not transmitted to the server
  - No proxy server!

https://docs.microsoft.com/en-us/iis/configuration/system.webserver/security/authentication/

Authentication

- .NET Passport Authentication
  - Permits single sign-on security (SSO)
    - User must not sign-in to every website separately
    - Example: EhB (office 365, cas, ibamaflex, canvas,...)
  - .NET Passport central server
    - Does not authorize or deny user access
    - It is the responsibility of the web site to control user permissions
  - Requests to IIS must contain valid .NET Passport credentials
    - If IIS does not detect .NET Passport credentials, requests are redirected to the .NET Passport logon page.

Authentication

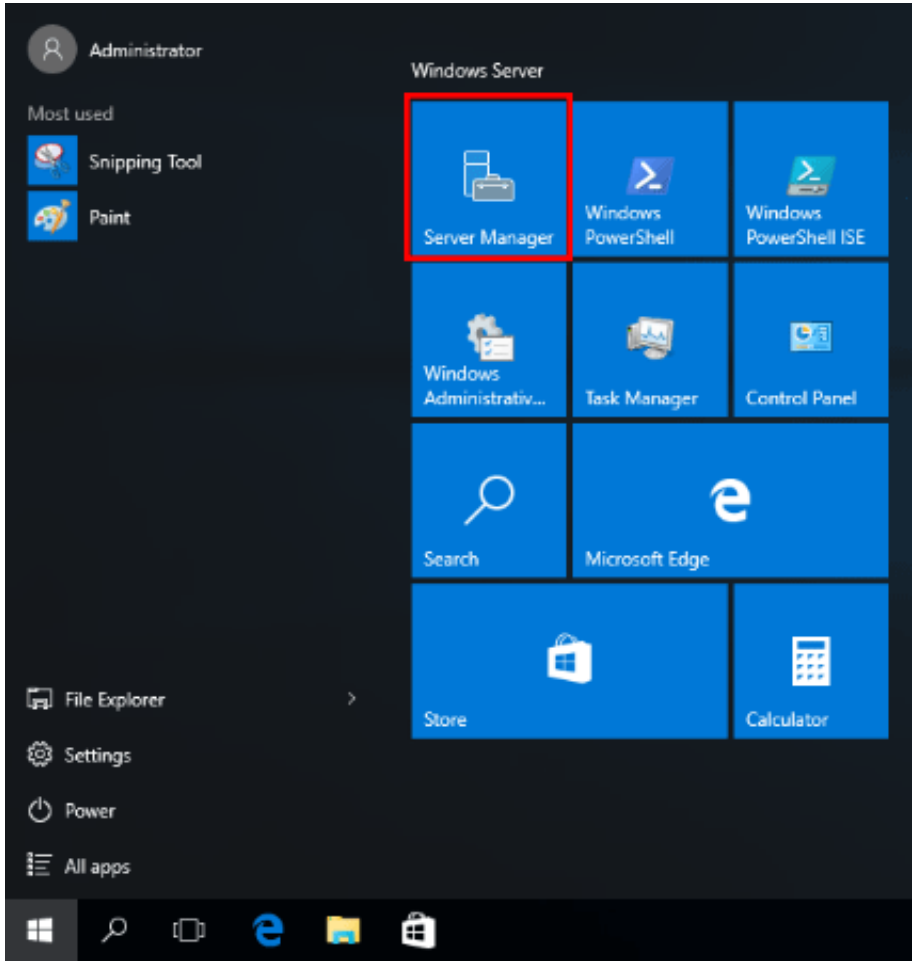
- Client Certificate Mapping
  - A mapping is created between a certificate and a user account
  - A user presents a certificate and the system looks at the mapping to determine which user account should be logged on
  - Mapping:
    - By using Active Directory
    - By using rules that are defined in IIS

Authentication

- You can configure each authentication method to control access to the following items on the IIS server:
  - All web content that is hosted on the IIS server
  - Individual web sites that are hosted on the IIS server
  - Individual virtual directories or physical directories that are in a web site
  - Individual pages or files that are in a web site

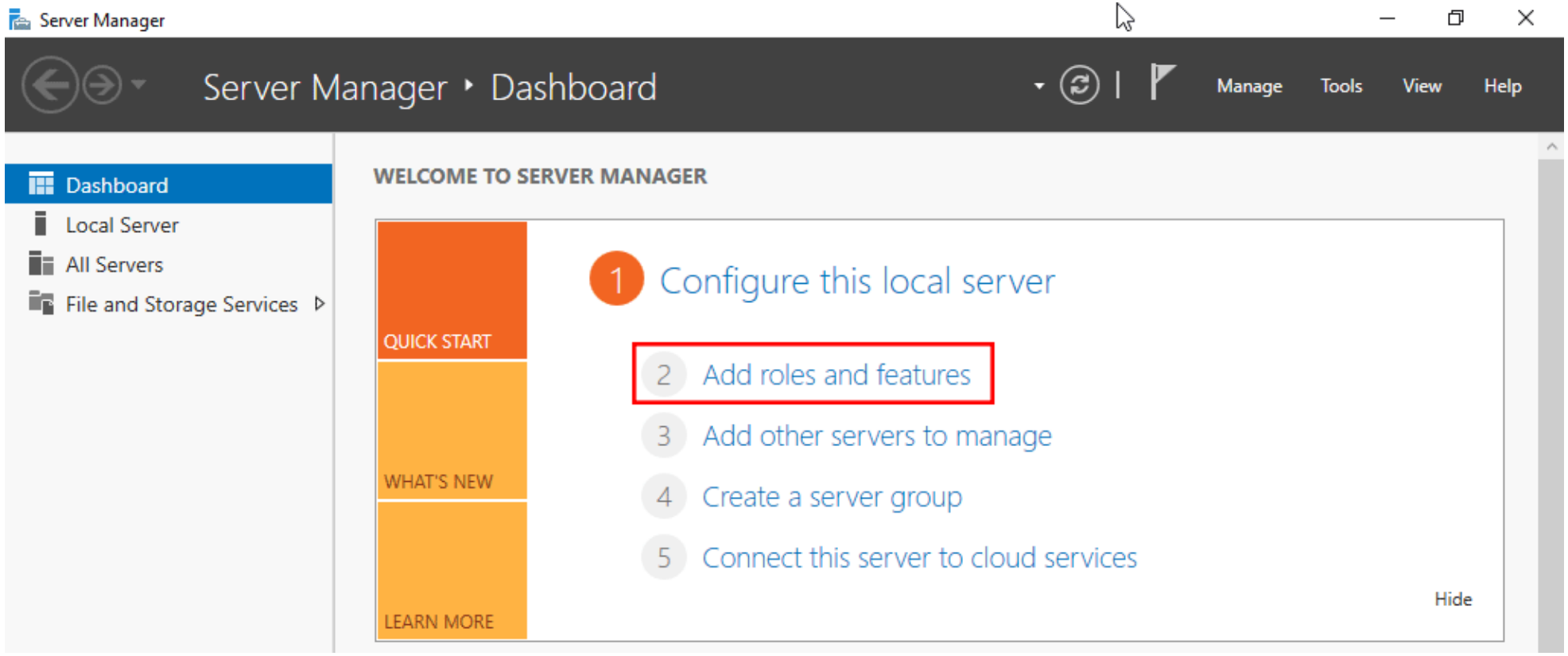
Installation

https://www.rootusers.com/how-to-install-iis-in-windows-server-2016/



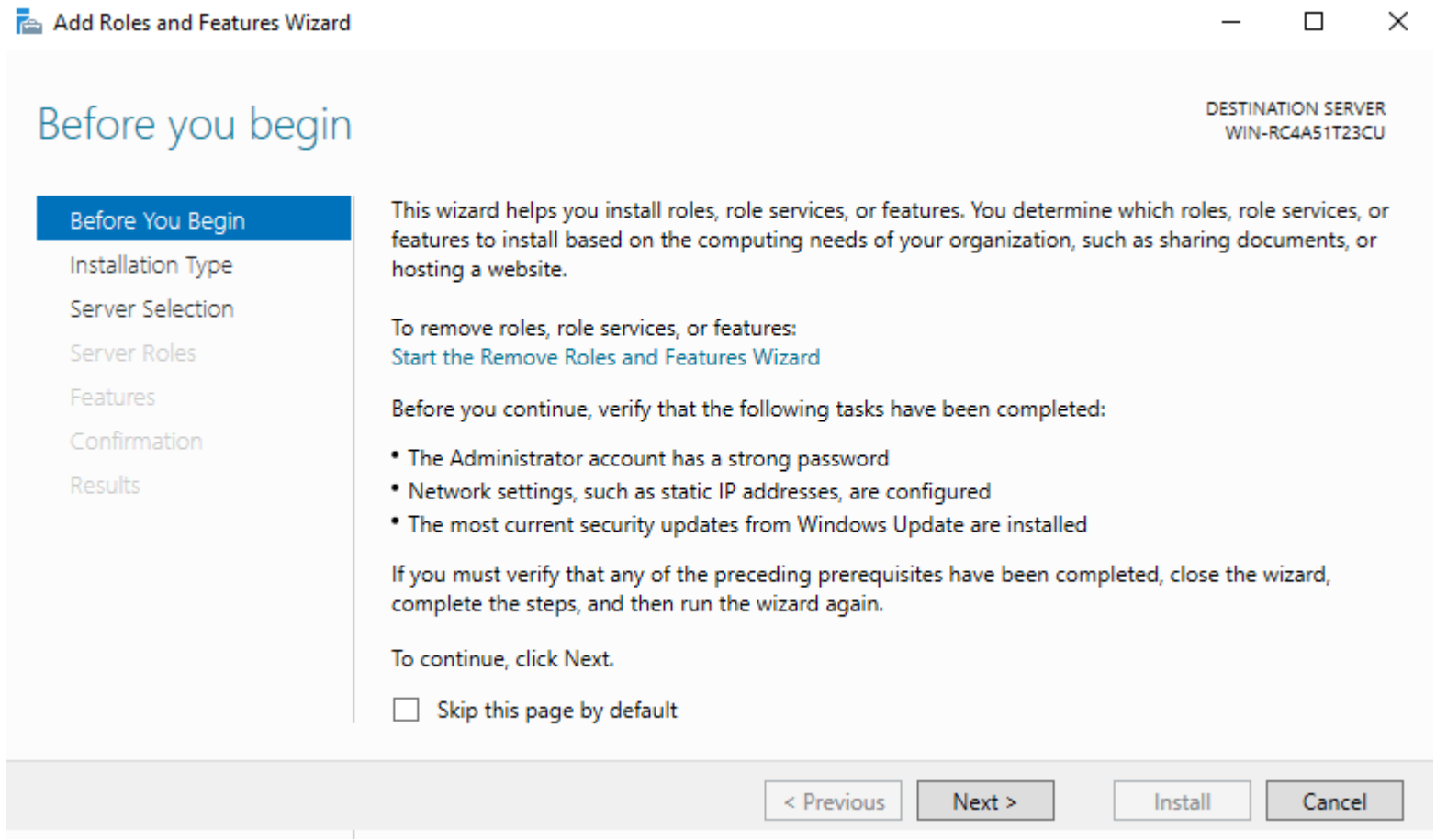
Open Server Manager

Installation



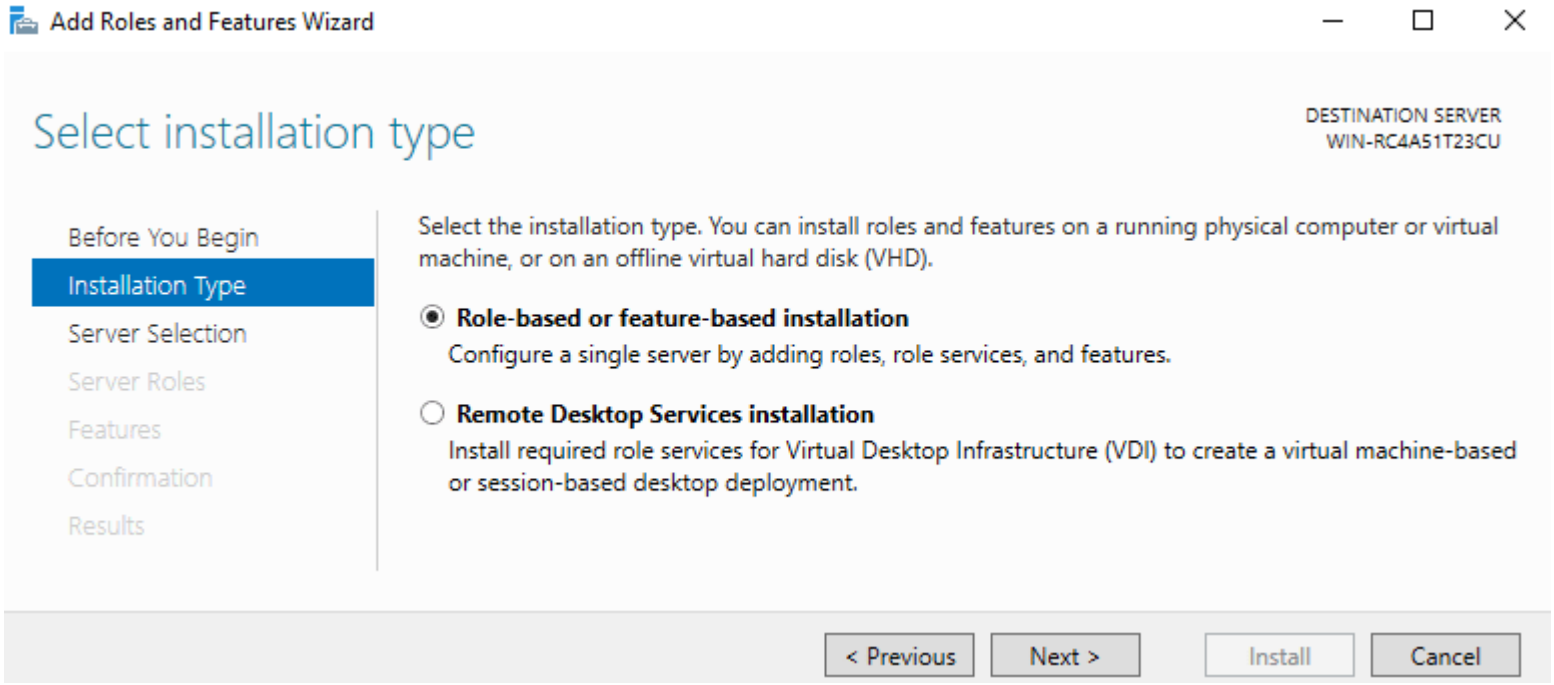
Add roles and features

Installation



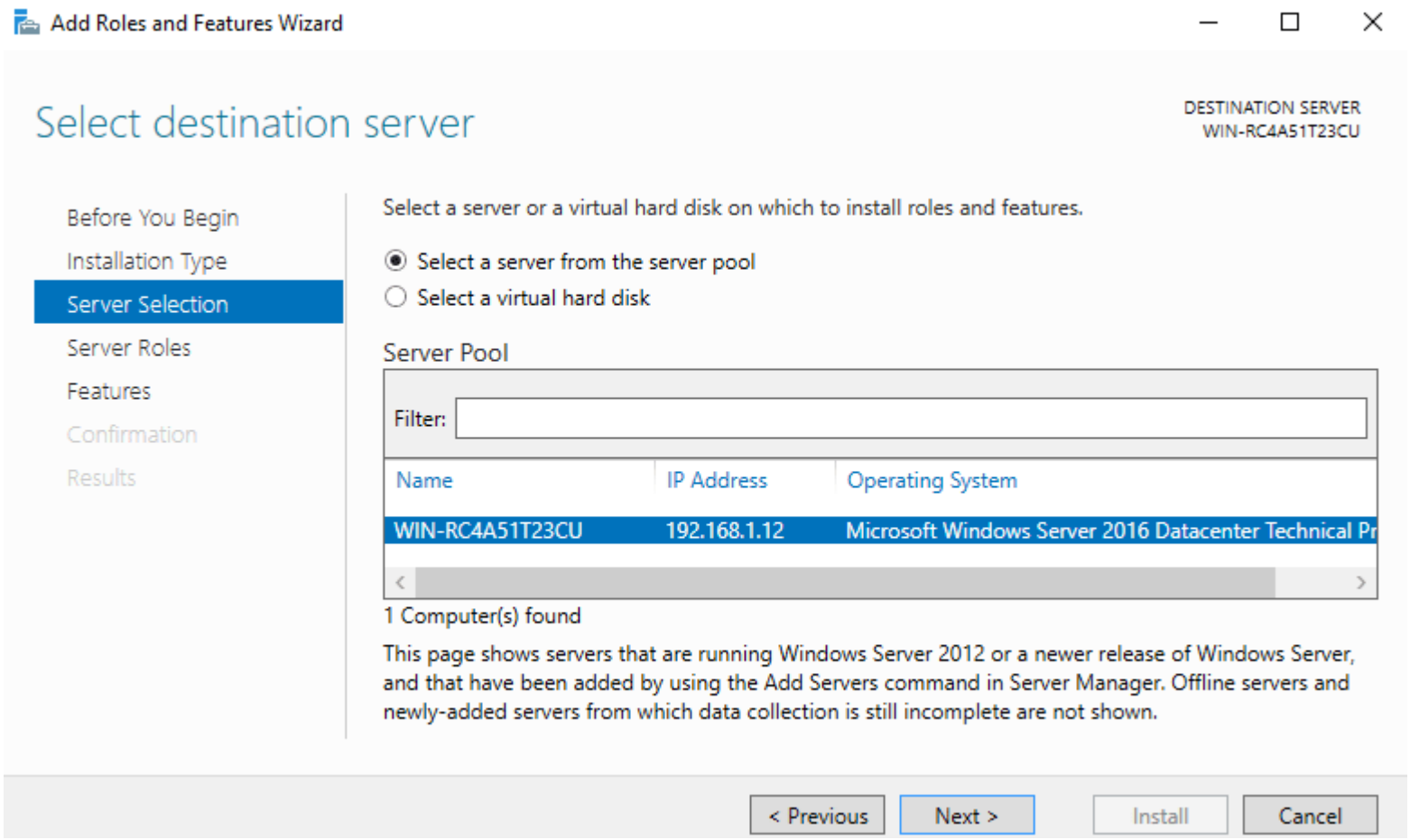
Click Next

Installation



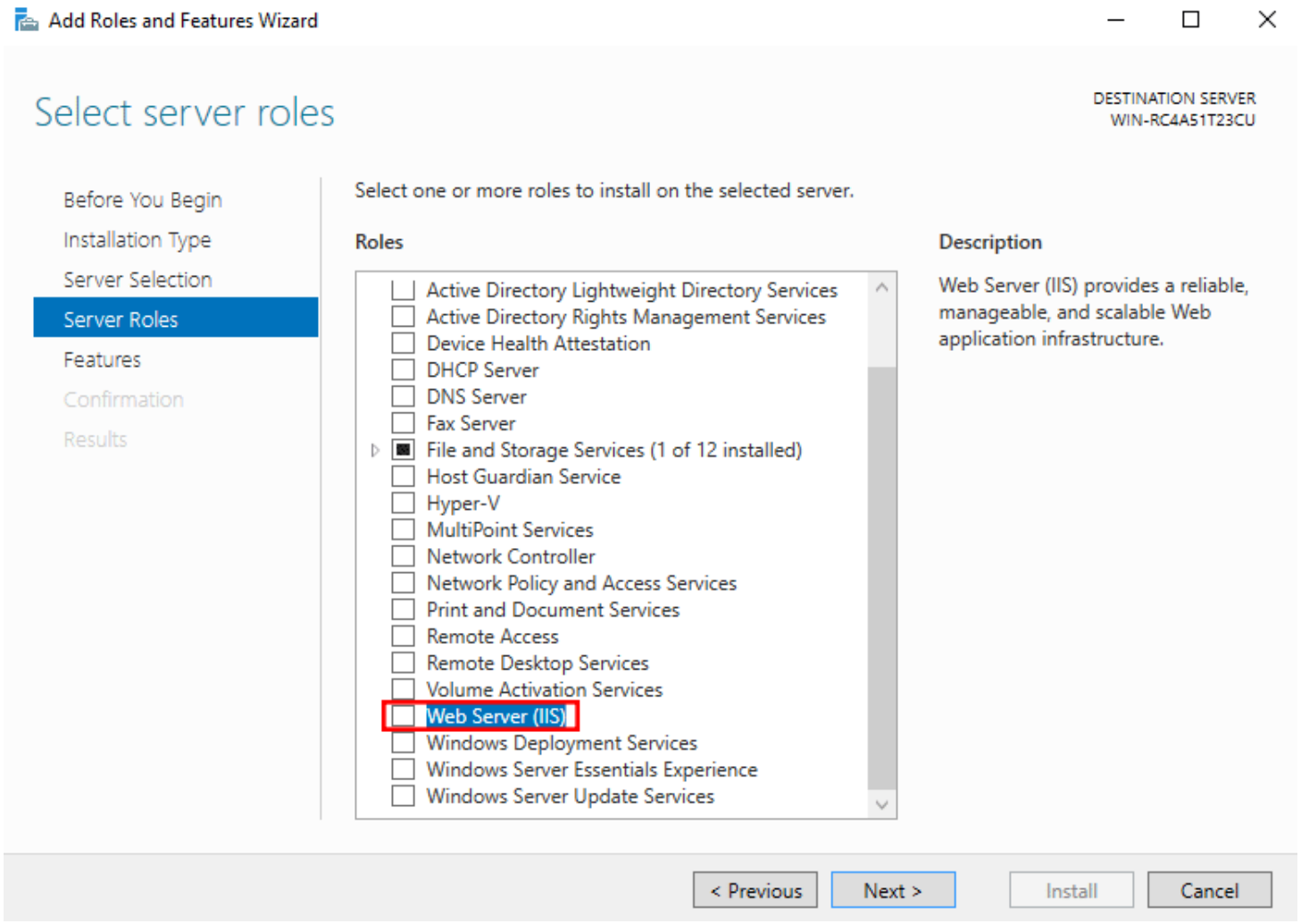
On the “Select installation type” window
→ leave “Role-based or feature-based installation” selected and click Next.

Installation



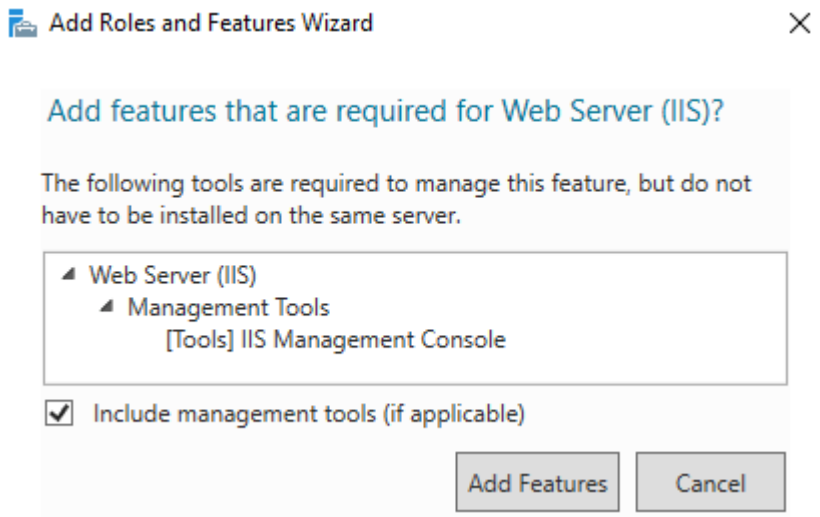
As we're installing to our local machine, leave "Select a server from the server pool" with the current machine selected and click Next. Alternatively you can select another server that you are managing from here, or a VHD.

Installation



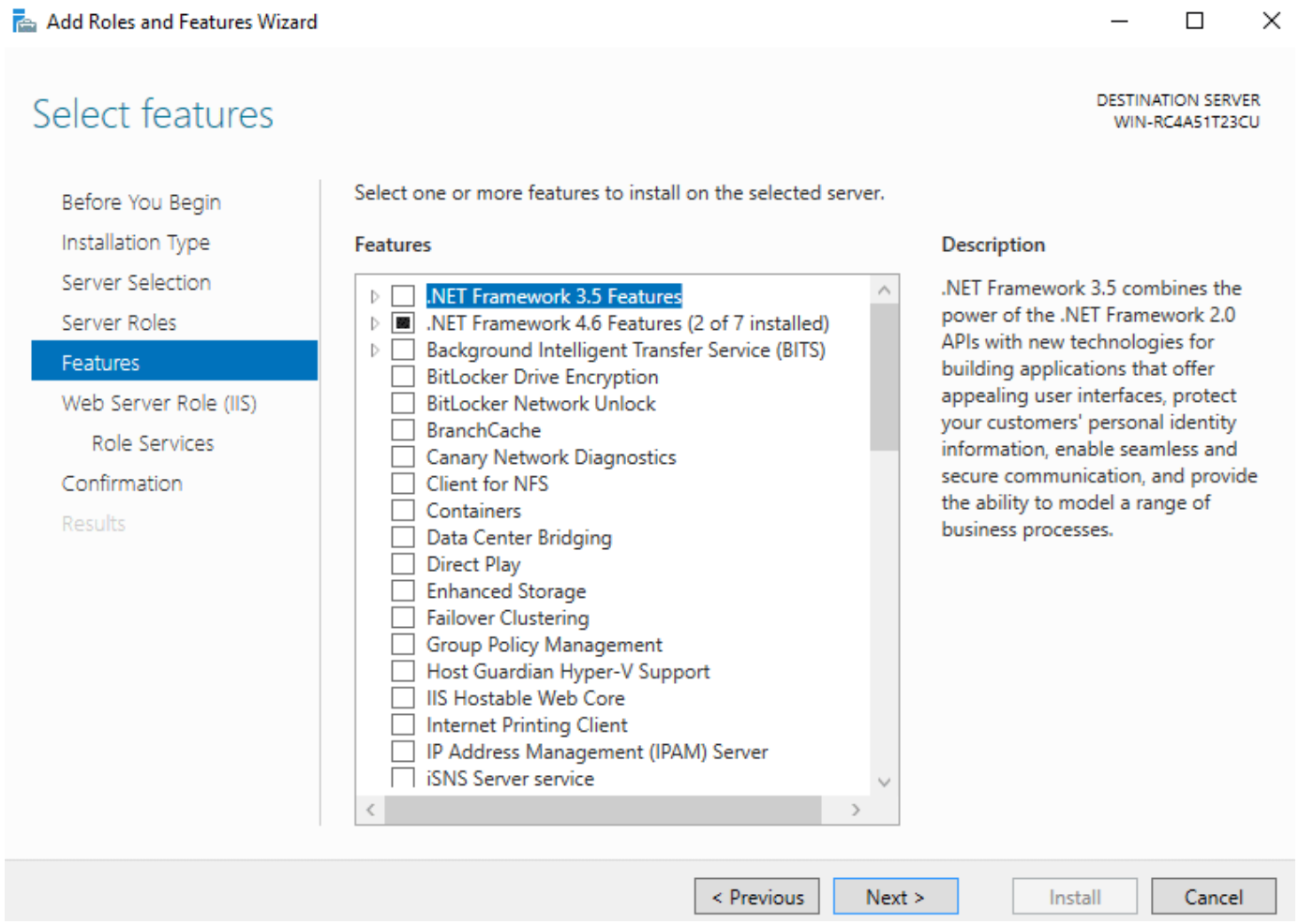
From the “Select server roles” window, check the box next to “Web Server (IIS)”. Doing this may open up a new window advising that additional features are required, simply click the “Add Features” button to install these as well. Click Next back on the Select server roles menu once this is complete.

Installation



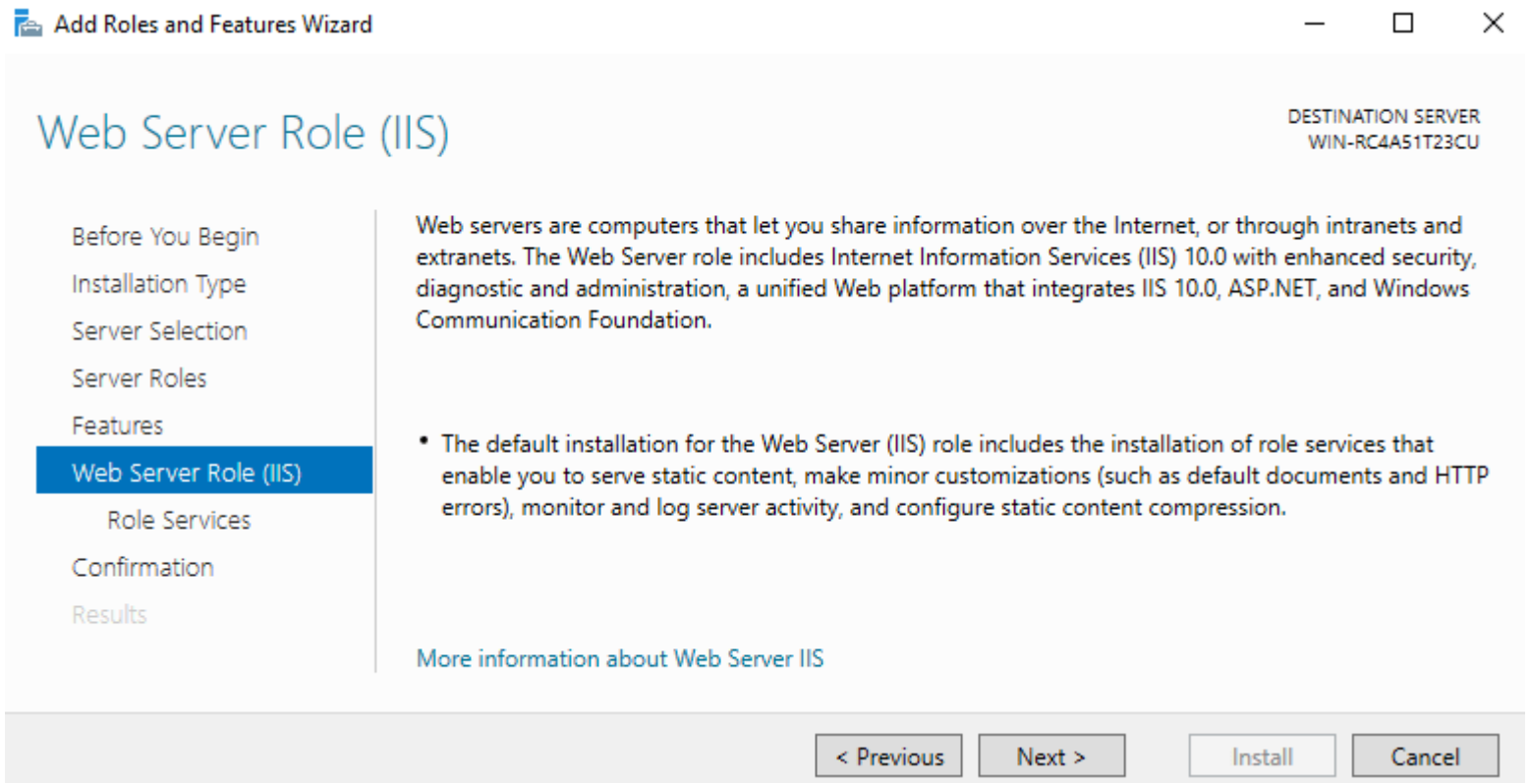
From the “Select server roles” window, check the box next to “Web Server (IIS)”. Doing this may open up a new window advising that additional features are required, simply click the “Add Features” button to install these as well. Click Next back on the Select server roles menu once this is complete.

Installation



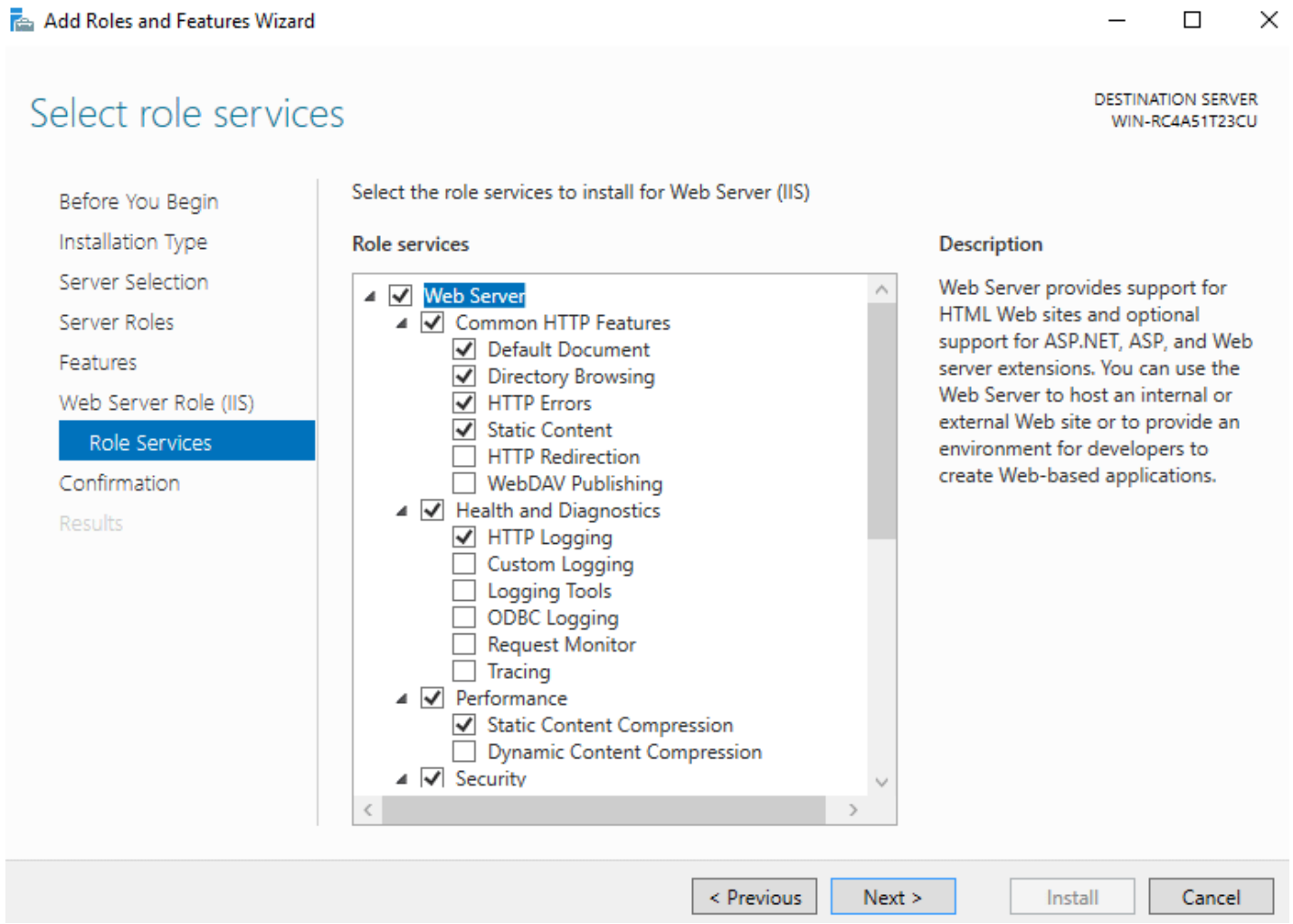
We will not be installing any additional features at this stage, so simply click Next on the “Select features” window.

Installation



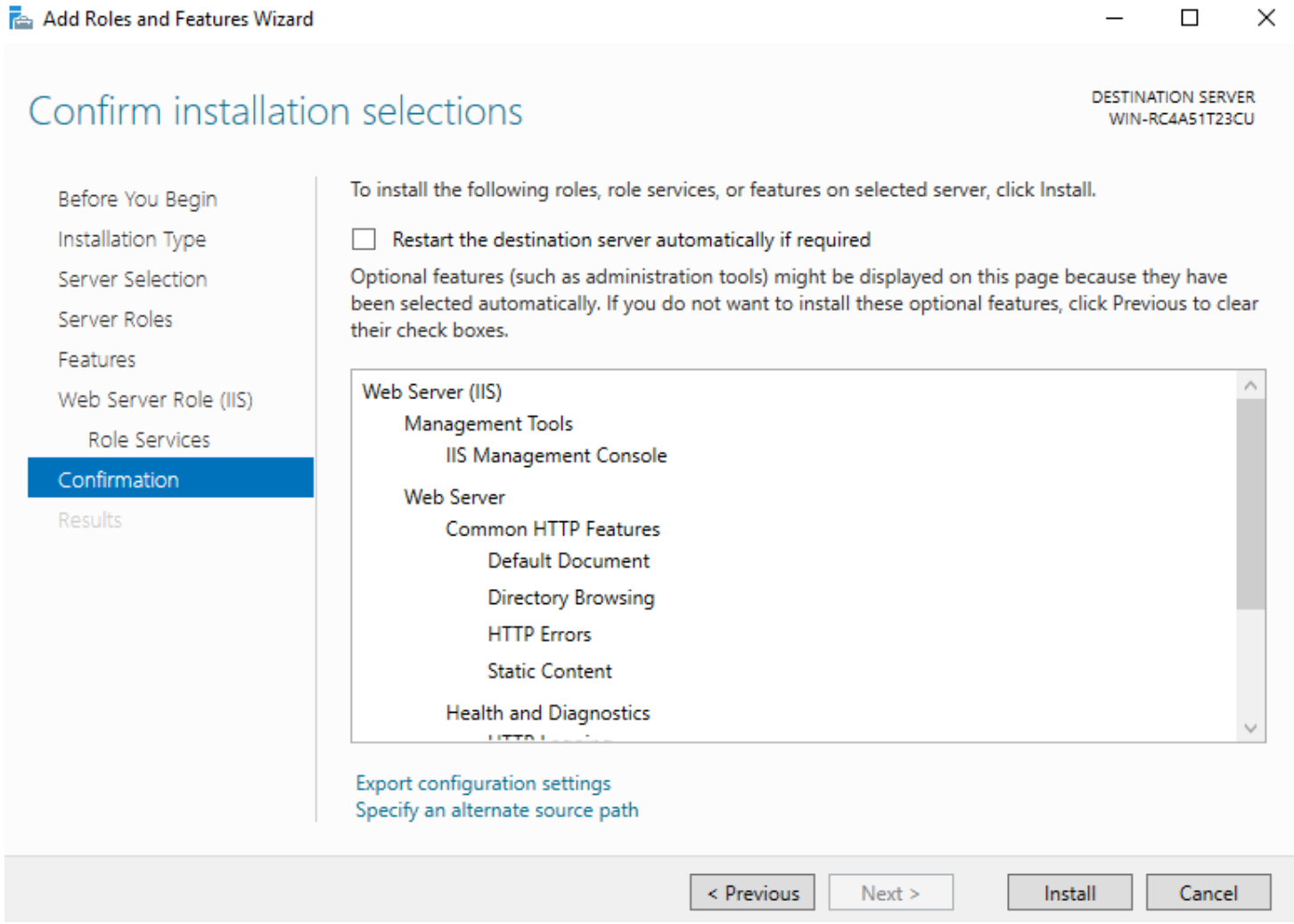
Click Next on the “Web Server Role (IIS)” window after reading the information provided.

Installation



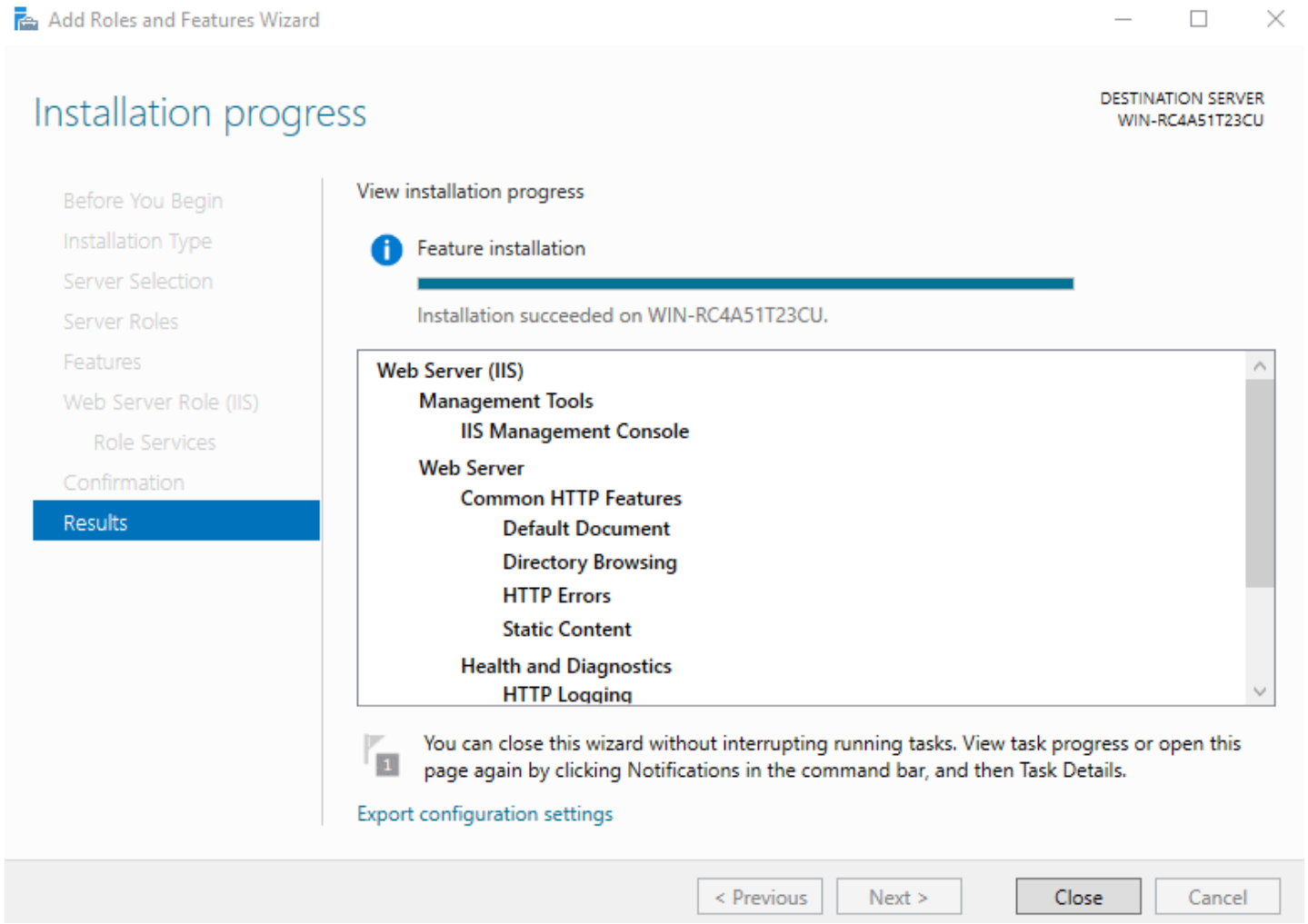
At this point on the “Select role services” window you can install additional services for IIS if required.
You don’t have to worry about this now as you can always come back and add more later, so just click Next for now to install the defaults.

Installation



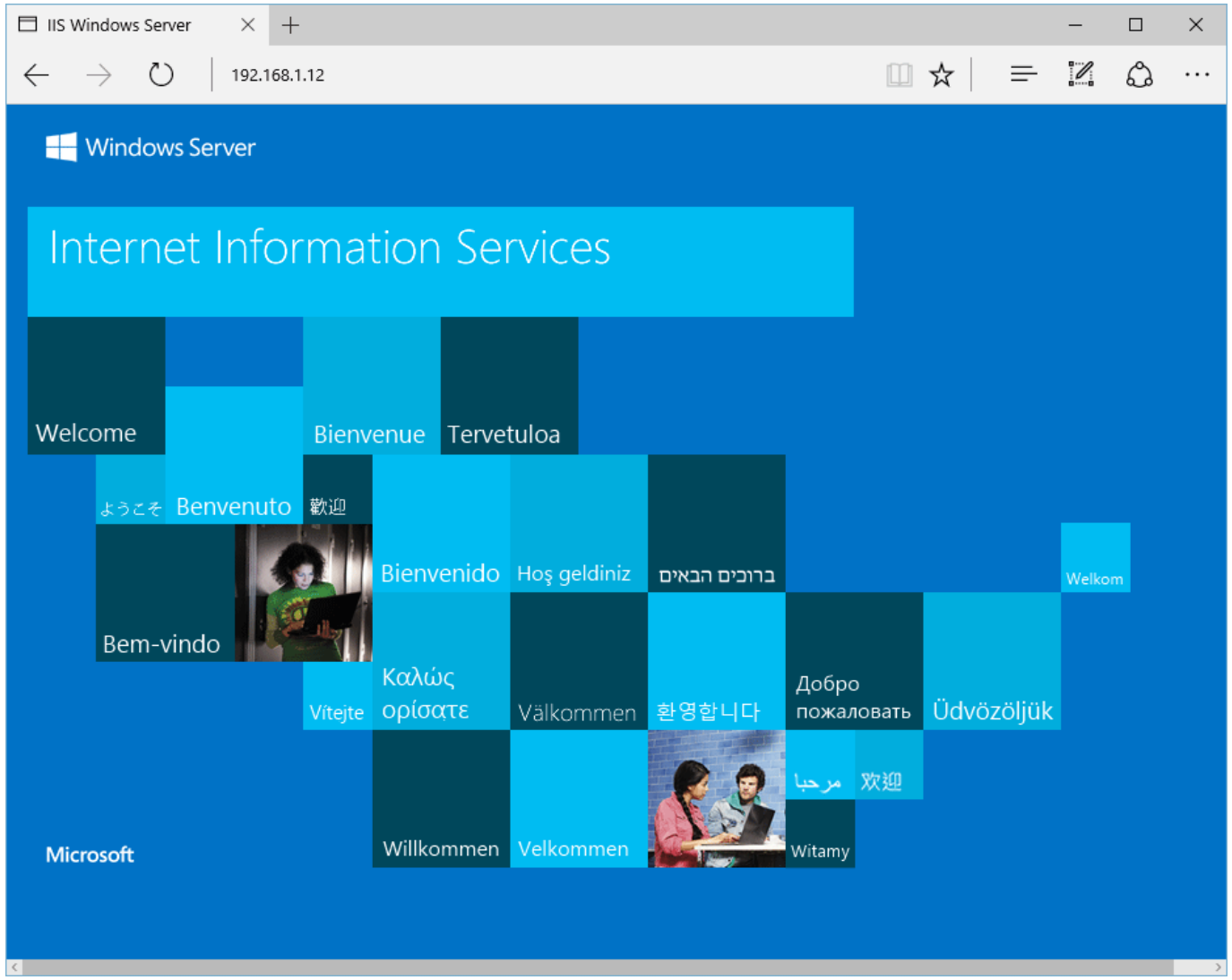
Finally on the “Confirm installation selections” window: Review the items that are to be installed and click Install when you’re ready to proceed with installing the IIS web server.

Installation



Once the installation has succeeded, click the close button. At this point IIS should be running on port 80 by default with the firewall rule “World Wide Web Services (HTTP Traffic-In)” enabled in Windows firewall automatically.

Installation



We can perform a simple test by opening up a web browser and browsing to the server that we have installed IIS on. You should see the default IIS page.

Installation

Or we can do all this by just executing one PowerShell cmdlet.

```
Install-WindowsFeature -name Web-Server -IncludeManagementTools
```