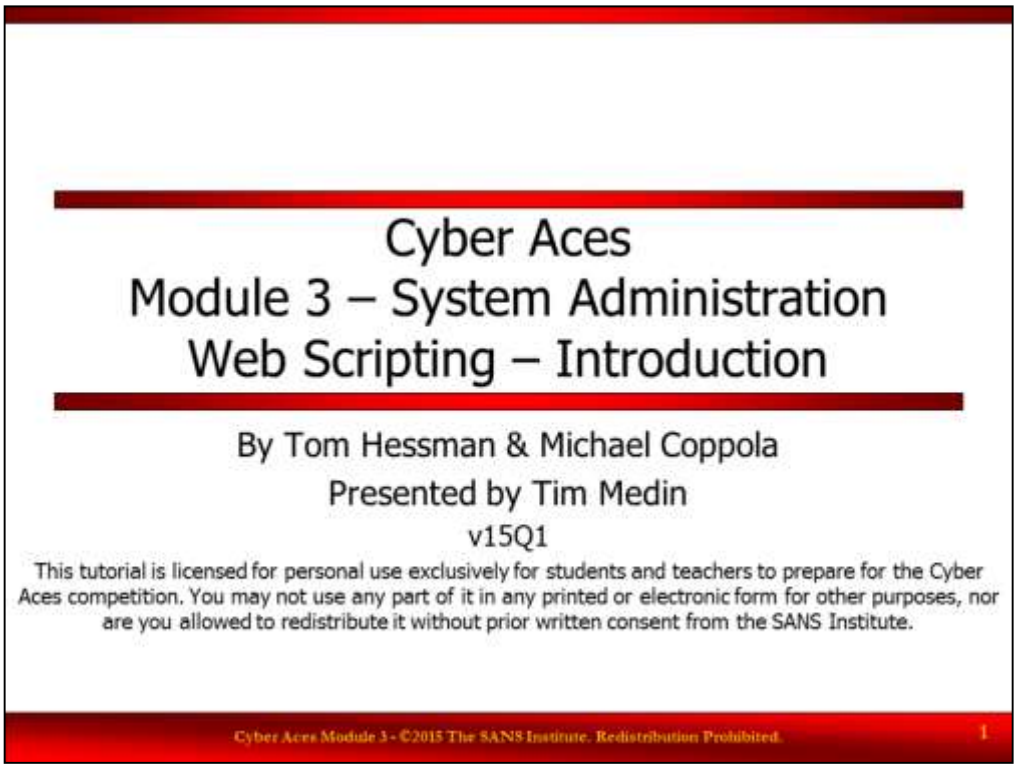
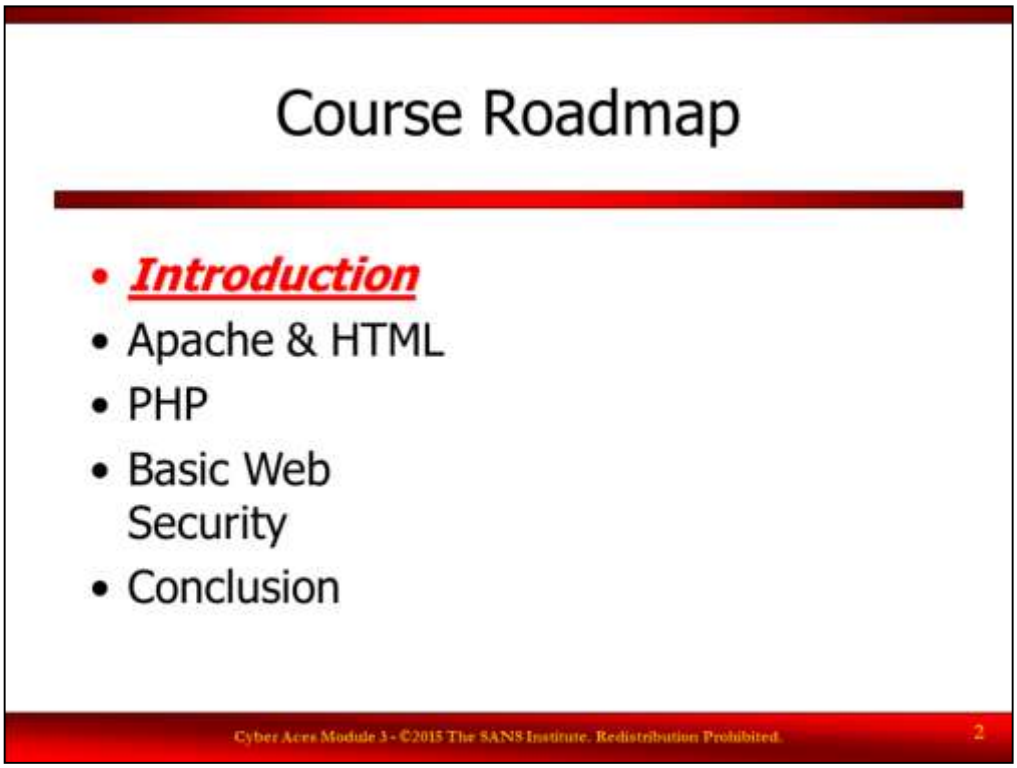
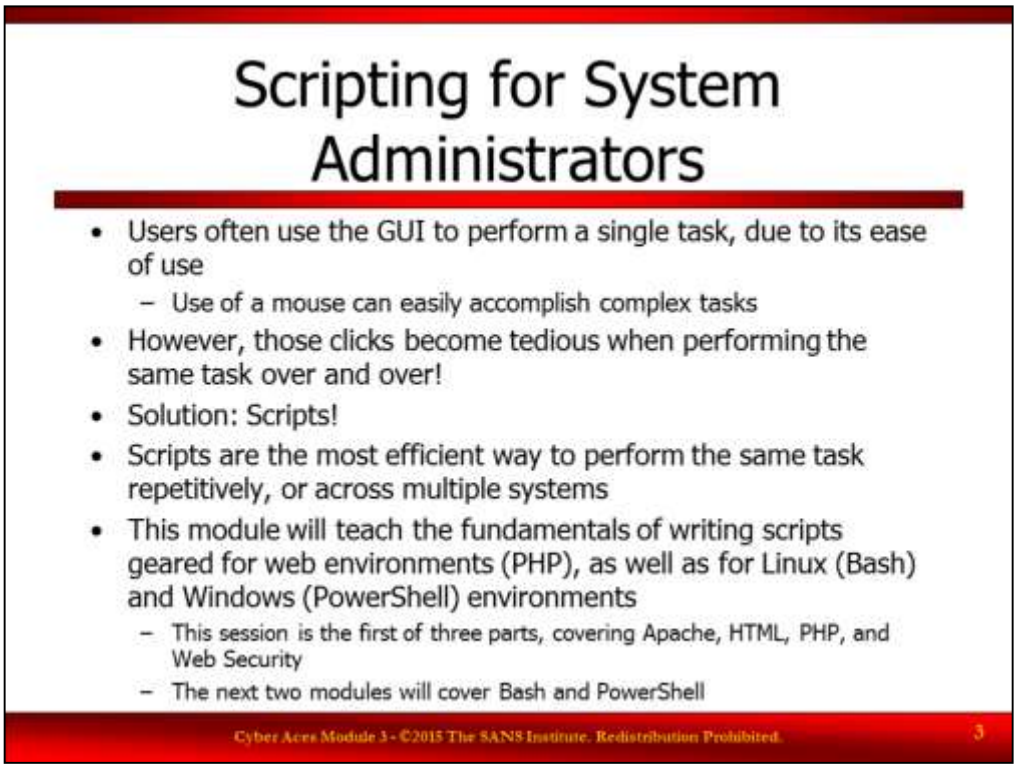


Welcome to Cyber Aces, Module 3! This module provides an introduction to the Apache Web Server, HTML, PHP, and basic web security.



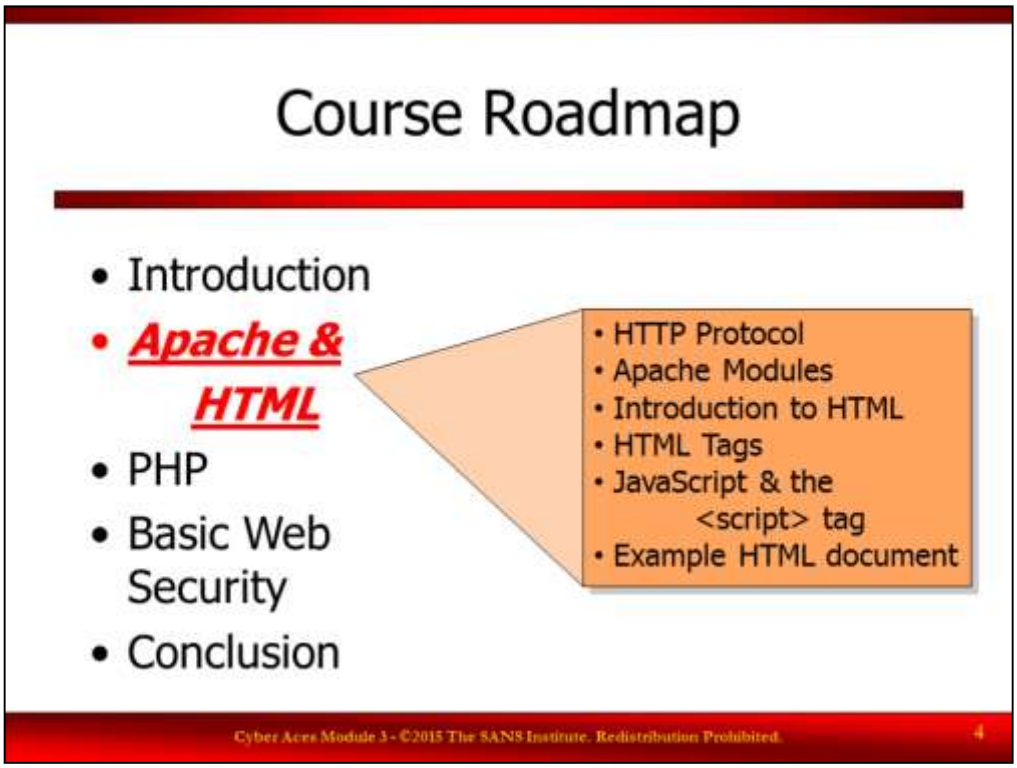
Course Roadap

Is this section, you will be introduced to Scripting for System Administrators.



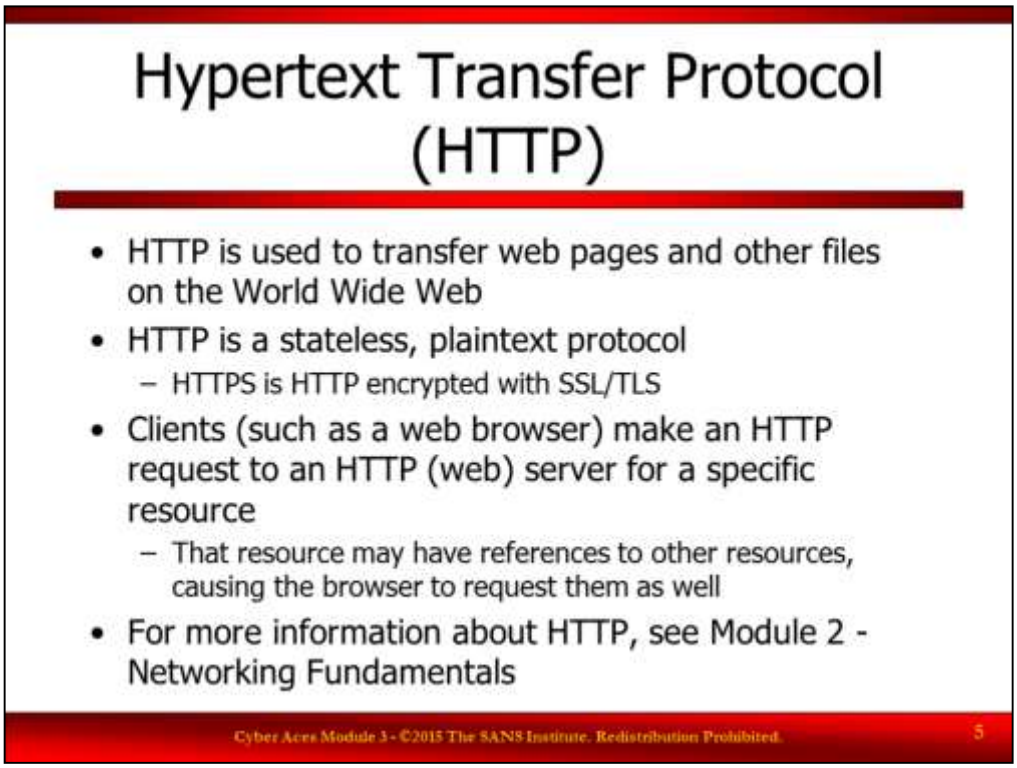
Scripting for System Administrators

Individuals who need to do a single task on a single machine will often use the GUI because of its ease of use. With just a few clicks of a mouse, you can easily accomplish complex tasks. However, those same clicks become tedious when the same task must be performed several times a day or performed on multiple computers. In those cases, resourceful IT professionals will often resort to developing command line scripts. A command line script is written once and can easily be run repeatedly, or be scheduled to run automatically by the computer. When you are supporting hundreds or thousands of computers, manually interacting with the programs becomes impractical and scripting is the only option. This course is intended to provide you with the tools you need to perform common administrative functions in some of the most popular scripting environments. We will examine PHP in the context of an Apache webserver, and then we will examine using GNU Bash and Microsoft PowerShell scripting from the command line to complete every day administrative functions.



Course Roadmap

In this section, we will quickly review the basics of the HTTP protocol, and then learn about the Apache Web Server.

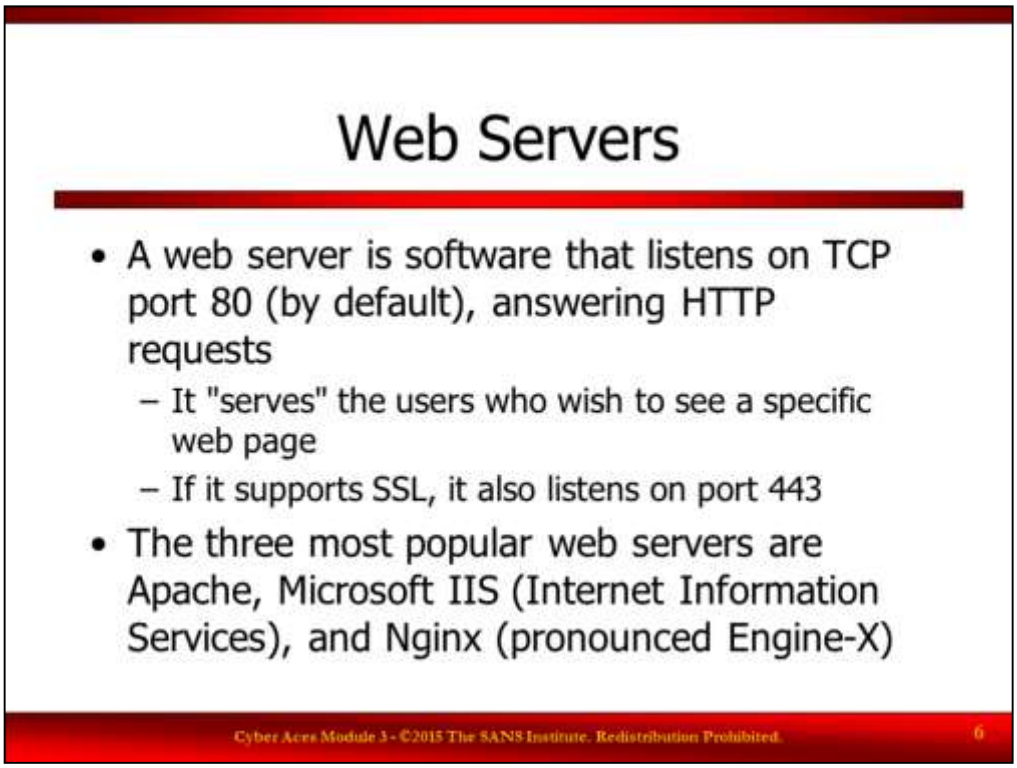


Hypertext Transfer Protocol (HTTP)

HTTP is used to transfer web pages and other files on the World Wide Web. It was originally designed to connect from a web browser to a web server and retrieve an HTML file, though its use and functionality has been greatly extended. HTTP is a stateless protocol, meaning that the client and server have no built-in way to maintain a session; each request stands on its own. HTTP is also a plaintext, almost human-readable protocol, making it unsuitable for sensitive data. HTTPS is the HTTP protocol encrypted using SSL/TLS encryption, which is used for secure transactions such as online banking.

Client software, such as a web browser, uses HTTP to make an HTTP request to an HTTP (web) server, asking for a specific resource (such as a particular file). The server then responds with that resource and the associated headers to create an HTTP response. That resource (such as an HTML file) may have references to other resources (such as images, scripts, or stylesheets), which the client will then make further requests for.

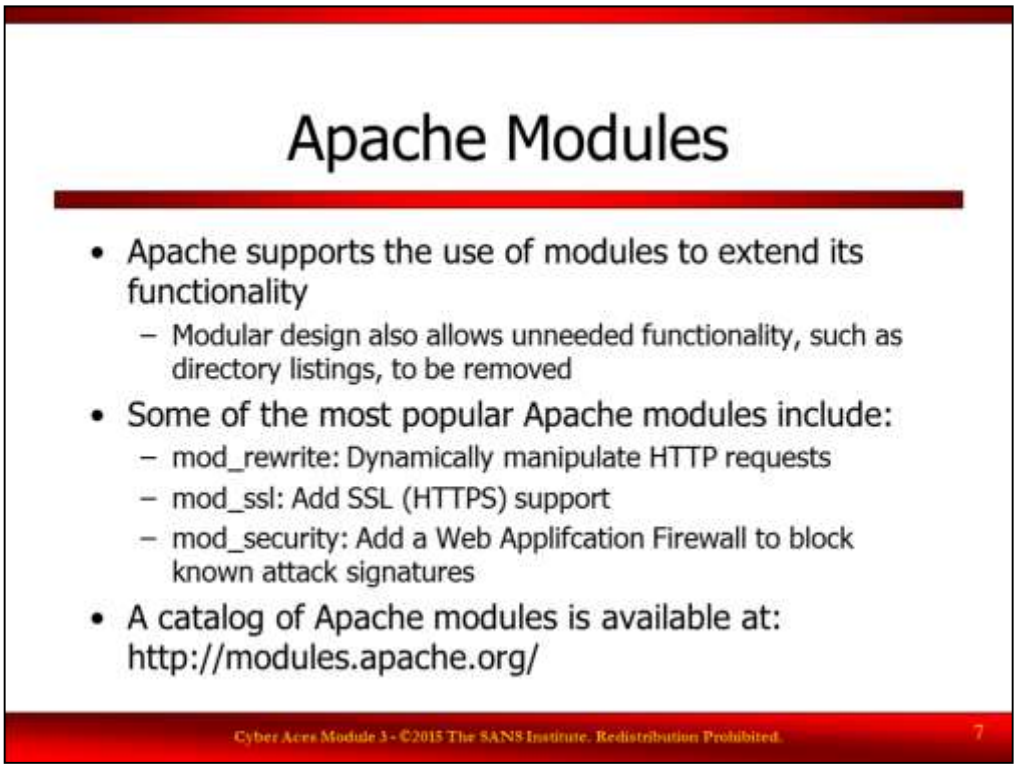
For more detailed information about HTTP, see Module 2 - Networking Fundamentals.



Web Servers

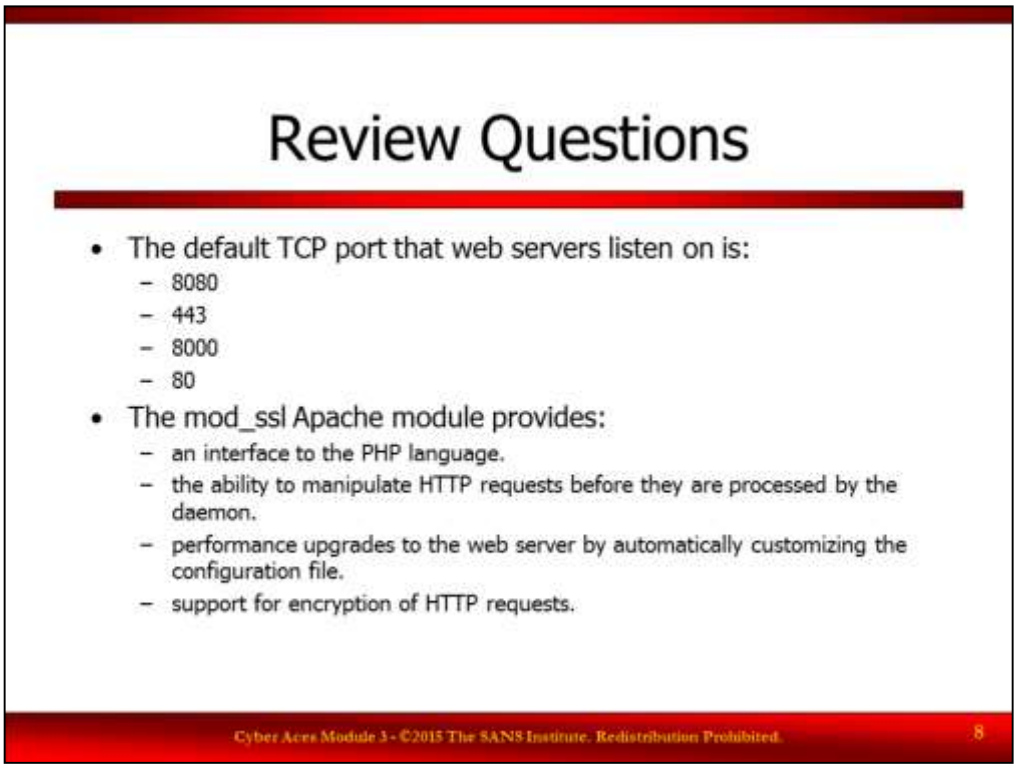
In order to access a website, the server that is hosting it must be running a specific piece of software called a web server. A web server listens on TCP port 80 by default, although this may be easily configured to a different port number. The purpose of a web server is to receive and respond to HTTP requests, in essence "serving" the users who wish to look at a specific web page through a web browser.

Although they all serve the same purpose, many different web servers exist. Different web servers offer the same core functionality, but some may also offer other features such as advanced load balancing, different kinds of authentication, support for different scripting languages (such as ASP, which is Microsoft-only), and support for HTTPS (encryption). The three most popular web servers are Apache, Microsoft IIS (Internet Information Services), and Nginx.



Apache Modules

System administrators may add greater functionality to their Apache installations through the use of modules. Popular modules include "mod_rewrite" (which lets you dynamically manipulate HTTP requests before they are processed by the daemon), "mod_ssl" (which adds support for encryption), and "mod_security" (which acts as an Intrusion Prevention System for Apache and attempts to block malicious requests). A catalog of Apache modules is available at http://modules.apache.org/.



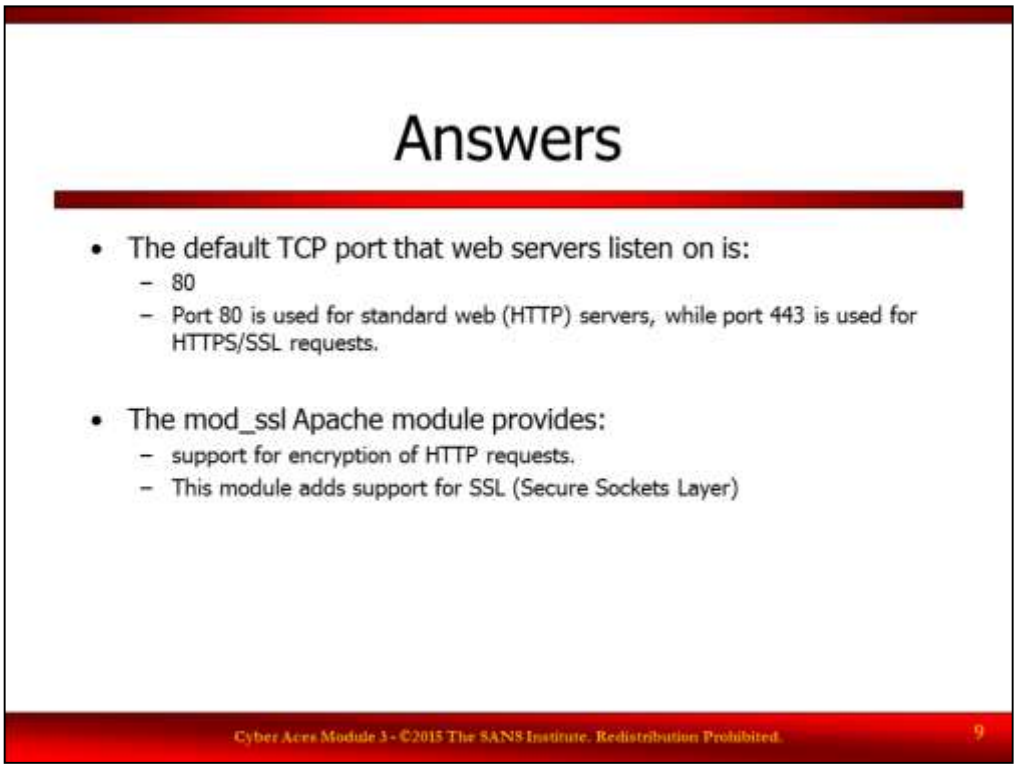
Review

The default TCP port that web servers listen on is:

- 8080
- 443
- 8000
- 80

The mod_ssl Apache module provides:

- an interface to the PHP language.
- the ability to manipulate HTTP requests before they are processed by the daemon.
- performance upgrades to the web server by automatically customizing the configuration file.
- support for encryption of HTTP requests.



The default TCP port that web servers listen on is:

Answer: 80

Port 80 is used for standard web (HTTP) servers, while port 443 is used for HTTPS/SSL requests.

The mod_ssl Apache module provides:

Answer: support for encryption of HTTP requests.

This module adds support for SSL (Secure Sockets Layer)

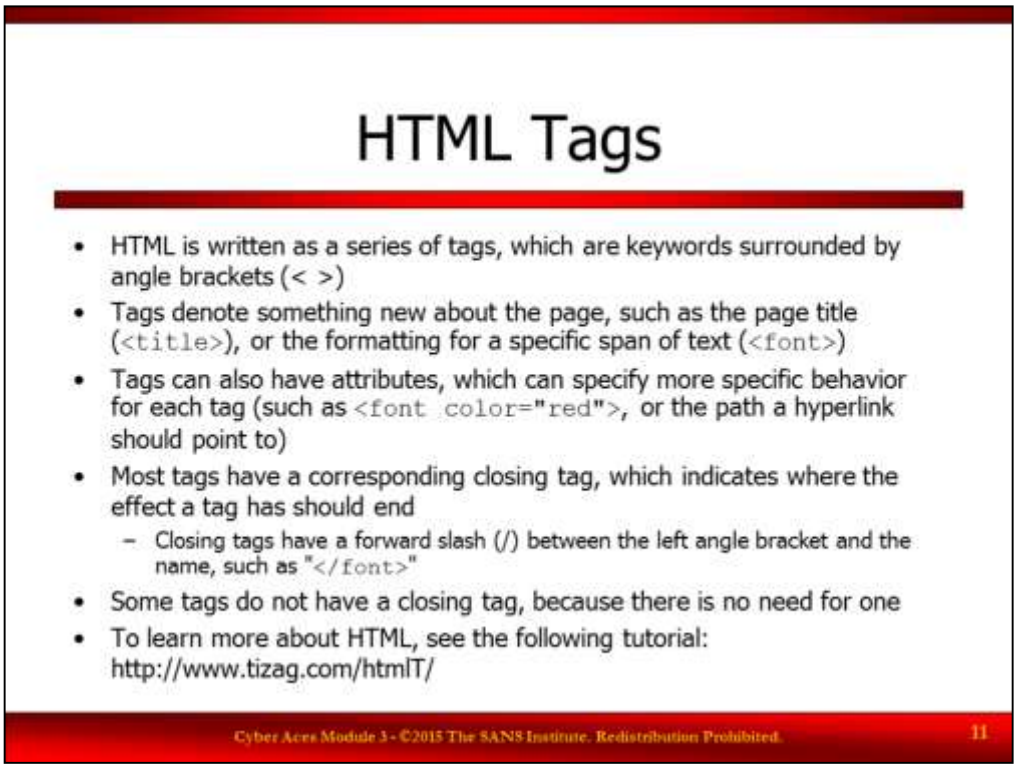
Introduction to HTML

When you view a web page, what are you seeing? There is most likely some text and possibly a few images. There are probably links to other pages, and sometimes forms that you can submit. How does your web browser know what to display on the screen? Why does it format text in a certain way or align an image to the right side of your screen instead of the left? The answer is HTML, which is an acronym for "Hypertext Markup Language." HTML is, in simple terms, code that your web browser interprets to display a web page. A web server will respond to an HTTP request with the HTML code of the requested page, and the web browser will display the page depending on the code it receives. Although HTML is a language, it should not be confused with a programming language such as PHP, Python, or C. It is instead a "markup language," which means that it is used to organize, format, and denote special characteristics of text. Thus, markup languages are not executed but instead parsed and interpreted. Programming languages are used to perform computations and calculate answers. Other markup languages such as XML also exist.

As an exercise, open up the website http://www.sans.org/. Once the page is loaded, right-click on an empty area of the page and select the option "View Source". The exact wording may be different depending on which web browser you are using. A new window should open with HTML in it. This is the code that your web browser processed in order to display the web page. By saving this code to your computer, you can then view the sans.org web page at any time, regardless of whether you are connected to the Internet or not. Note that the page loads other content too, such as images and JavaScript files (.js files, which we'll go over later), so the page will seem broken unless you also save all of these individual files. Additionally, any dynamic functionality of the page will be removed, since you are not able to see the PHP code that generated the HTML, but only the HTML itself. Maybe a bookmark would be better instead!

It should be noted that the sans.org website is very complicated and contains a lot of content. The HTML code is made up of many, many tags and you may see some content that we have not gone over how to use yet. As you become more familiar with HTML, you will be more comfortable looking at complex code like this.

In an effort to standardize how HTML is written, the World Wide Web Consortium (W3C) offers the Markup Validation Service at http://validator.w3.org/ to check markup validity of submitted code. To put it simply, it critiques the HTML code that people write and offers suggestions on how to make it more standard. By standardizing the code that you write, it has a better chance of working across all web browsers, since each one interprets a little differently. Code that renders perfectly on Firefox might be a mess on Internet Explorer, and vice versa. Even though it may take a little more time and effort to write standardized code, it saves a lot of headache later when you want to make it look the same for everyone viewing your website.



HTML Tags

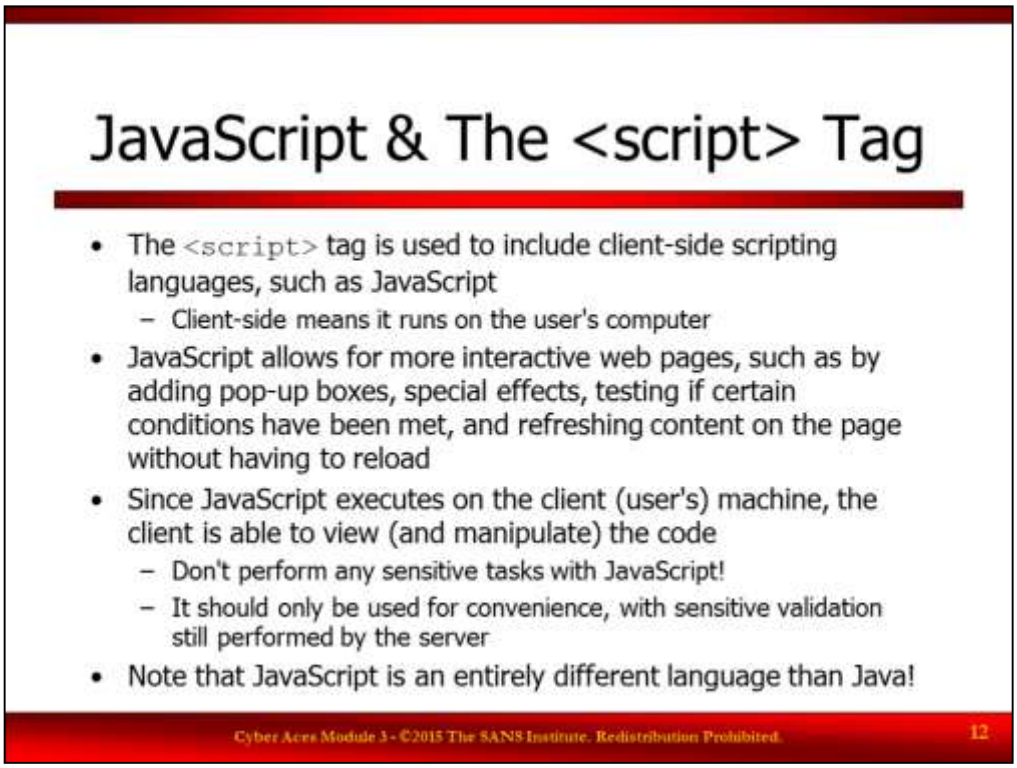
HTML code is written as a series of tags, or keywords in the language surrounded by angle brackets ("<" and ">"). When you write a tag, you are denoting something new about the page, such as the page's title (using the "<title>" tag) or formatting for a specific span of text (using the "" tag). You can also make tables (using the "<table>" tag), make links to other websites (using the "<a>" tag), and display images on the page (using the "<img>" tag). Within many tags are attributes, which give greater control over the specific behavior of each tag. For instance, the "" tag lets you decide which color the text should be with the color attribute, like so: . The "<img>" needs to know which image you want to display through the src attribute, so you would use it like: <img src="lolcats.jpg">.

In HTML, most tags also have a closing tag. That is, once you include a particular tag in your web page, you must also specify where the effects of that tag end. For instance, the "" tag is closed by the "" tag. All closing tags are the name of the opening tag with a forward slash in front of the keyword. For example:

```
<font color="blue">This text is blue.</font> <i>This text is not blue, but it is italicized.</i>
```

Some tags do not have a closing tag, such as "<img>" or "
", simply because they have no need for one. A web browser would have no further information to gain by finding the closing tag to an image or a line break, so the language combines them into single tags.

If you wish to learn more about HTML and the many different tags that are available for use, check out the tutorials at http://www.tizag.com/htmlT/. It is not necessary to memorize every tag, but as you gain experience and start building more complex websites, you will naturally begin to know many of the most common ones by heart. Select a few HTML tutorials from the list on Tizag (they are all very short and exemplify usage of the code) and see what interests you!



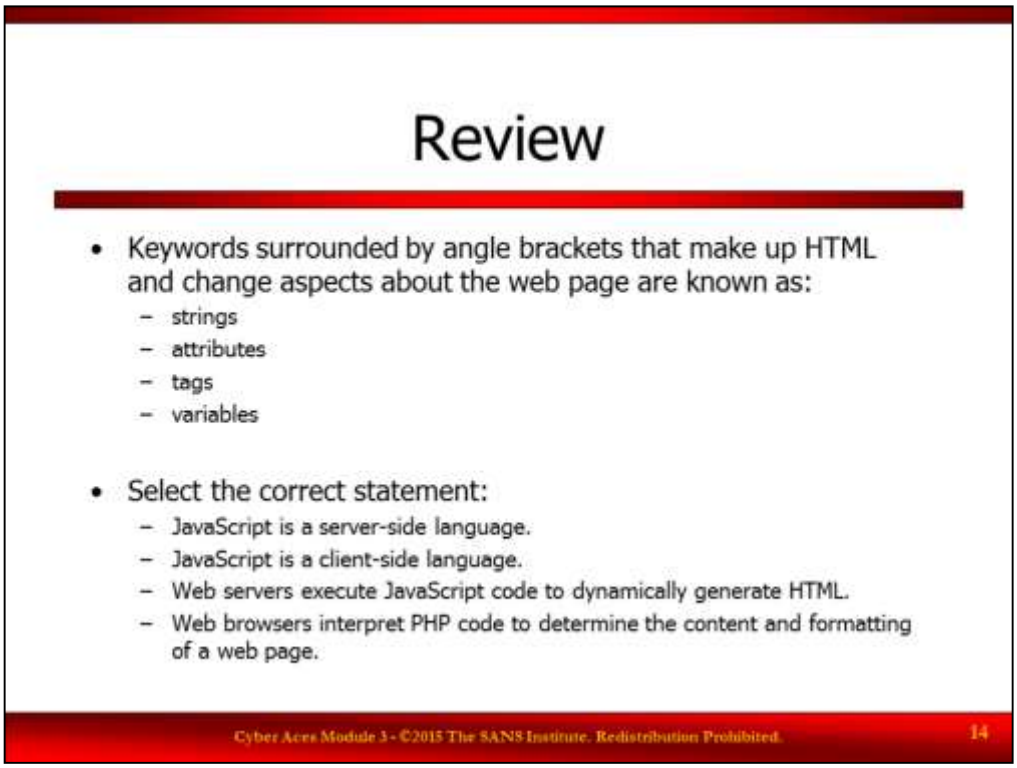
JavaScript & the <script> Tag

One special tag is the "`<script>`" tag, which lets you include JavaScript code within your HTML document. JavaScript is a scripting language that adds dynamic content to a web page, allowing you to create pop-up boxes, add special effects, and determine whether certain conditions have been met (like password strength when registering for a forum), and refreshing content on the page without having to reload the page. JavaScript, unlike PHP, is client-side, which means that the user's web browser is executing it and not the server. Because of this, you are able to see all of the JavaScript code in the page, whereas PHP code is hidden on the server. This means that while JavaScript introduces a large amount of functionality to websites, it shouldn't be used for handling sensitive processes since the client (you) can see (and potentially manipulate) everything that happens! If everyone were able to see the code for a login panel, then it would be a huge security concern!

Note that JavaScript is an entirely different language than Java! Java can be used on the server side, in the form of servlets or JSP (Java Server Pages), or on the client side (in the form of applets). Java requires a separate browser plug-in to run applets, whereas most web browsers have built-in support for JavaScript.

Example HTML Document

Here is a very simple example of an HTML document, along with a screenshot of how it looks in a web browser. The `<title>` tag defines the document title that appears in the browser's title bar and in the current tab (neither of which are visible above). The `<a>` (anchor) tag defines a hyperlink, and the "href" (horizontal reference) attribute specifies where the hyperlink should point. The `<font>` tag changes the style of the text enclosed within it, with the "color" attribute specifying to make the text red.



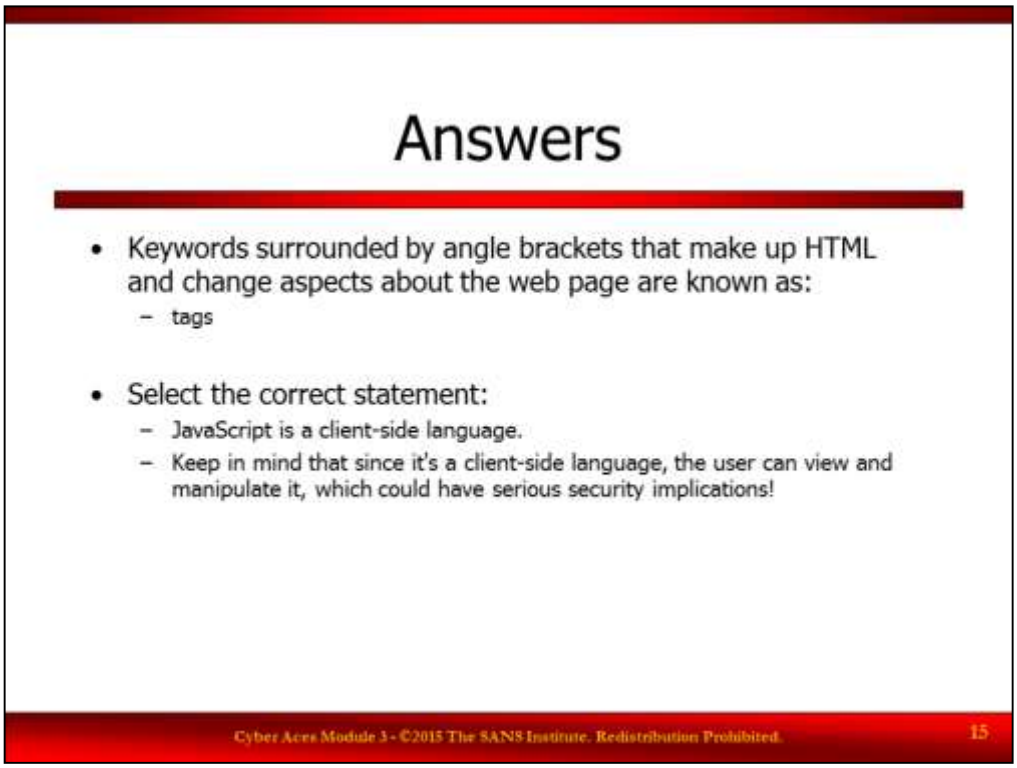
Review

Keywords surrounded by angle brackets that make up HTML and change aspects about the web page are known as:

- strings
- attributes
- tags
- variables

Select the correct statement:

- JavaScript is a server-side language.
- JavaScript is a client-side language.
- Web servers execute JavaScript code to dynamically generate HTML.
- Web browsers interpret PHP code to determine the content and formatting of a web page.



Answers

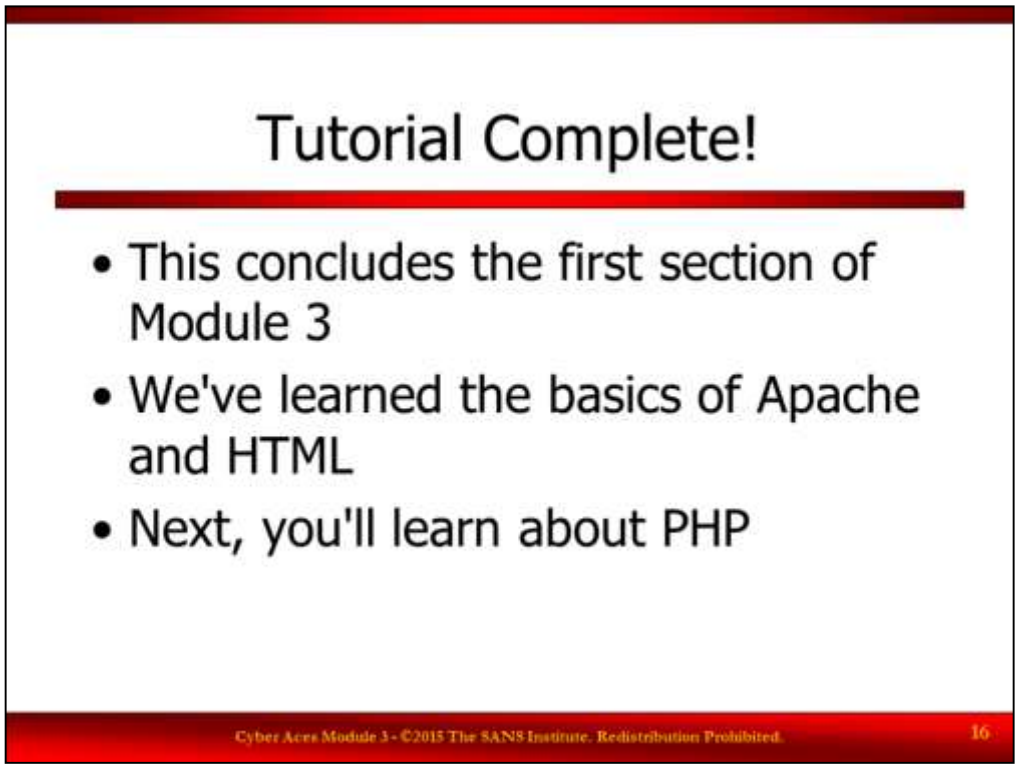
Keywords surrounded by angle brackets that make up HTML and change aspects about the web page are known as:

Answer: tags

Select the correct statement:

Answer: JavaScript is a client-side language.

Keep in mind that since it's a client-side language, the user can view and manipulate it, which could have serious security implications!



Tutorial Complete

This concludes the first section of Module 3. We've learned the basics of Apache and HTML. Next, we'll learn about PHP.