



SSL in the e-Commerce Framework

SSL and the PublicStore Web.config

Setting up SSL to work with a site created by the e-Commerce Framework is an easy task. Of course, you must first be sure to purchase a SSL Certificate. A quick search using the key words “SSL Certificate” in google will reveal many places to acquire one for your site. Instructions on how to set up your SSL Certificate should be given to you from the seller of the Certificate. After purchasing and setting up your SSL Certificate you can then proceed to the PublicStore web.config file.

SSL requires that absolute URL paths be used when switching protocols from page to page. This causes a problem when a visitor to your site clicks on a link that should bring them to a secure page – the reference must be absolute in order to switch protocols from HTTP to HTTPS. The problem with requiring absolute URLs is obvious – what if, for example, your domain name changes? You will have to go back through all your links which were absolute and correct all of them.

SSL and the PublicStore Web.config

This is where the SecureWebPageModule comes into play and is implemented in the Web.Config of the public store. The SecureWebPageModule allows a user to specify files and directories to make secure without the need for specifying an absolute url. Below is a sampling of code from our web.config file in the PublicStore directory:

```
1 <secureWebPages mode="Off">
2   <file path="/profile/login.aspx">
3   </file>
4   <directory path="/checkout">
5   </directory>
6 </secureWebPages>
7 <location path="Profile/Login.aspx">
8   <system.web>
9     <authorization>
10      <allow users="?" />
11      <deny users="*" />
12    </authorization>
13  </system.web>
14 </location>
15 <location path="Profile/Logout.aspx">
16   <system.web>
17     <authorization>
18      <deny users="?" />
19    </authorization>
20  </system.web>
21 </location>
22 <location path="Profile/OrderHistory.aspx">
23   <system.web>
24     <authorization>
25      <deny users="?" />
26    </authorization>
```

Partial listing of the PublicStore web.config file
Figure 1

What you can gather from the above section is that you must turn on secureWebPages mode to enable security. Only those pages within the <secureWebPages> </secureWebPages> tags will be using the HTTPS protocol.

You can specify files which you desire to be fetched using the HTTPS protocol as demonstrated by the second and third lines in our figure 1 listing. Also, you can set entire directories to be served up in HTTPS as demonstrated above as well in lines 4-5.

A third thing you may notice is that you can set authorization to either allow or disallow users to view a certain page. For example, in the above code in lines 7 through 14, authorization is set for the login page. In this case we allow anonymous users to see the login page while we deny all other users whom are authenticated users (since users whom are already logged in, have no need to see the login page).

Another example, not shown above:

```
<location path="Profile/OrderHistory.aspx">  
  <system.web>  
    <authorization>  
      <deny users="?" />  
    </authorization>  
  </system.web>  
</location>
```

Section of the PublicStore web.config file

Figure 2

In this example we are denying anonymous users the right to see the order history page, in other words, one must be authenticated.

Conclusion

That is SSL and the ECF in a nutshell! You can have your site secure in 3 easy steps

- 1) Acquire a digital certificate
- 2) Set up your digital certificate
- 3) Configure your security in the PublicStore web.config file

For extra information checkout the comments above the web page security section in the PublicStore web.config file. Therein are short, straight to the point instructions on how set up your security.