NOMADIX®

CAPTIVE PORTAL AND THE NEW SECURITY PARADIGM

OPTIONS FOR HANDLING REDIRECTION PROBLEMS CAUSED BY CERTIFICATE MISMATCHES

30851 Agoura Road, Suite 102
Agoura Hills, CA 91301
818-597-1500 Main
818-575-2480 Sales
www.nomadix.com

INTRODUCTION

Over the past year, more and more major websites have begun using SSL (Secure Socket Layer or "https://") to ensure a secure connection with their servers. While using SSL has long been a standard with banks and other organizations that transfer sensitive information online, now companies like Facebook, Google and Yahoo are following suit. Since many users make sites like Google and Yahoo their homepage, this poses a unique challenge for operators of guest networks and hotspots.

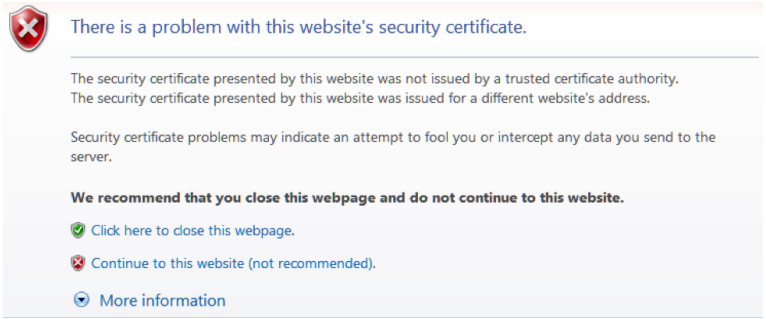
The increased security that results from using SSL benefits users because their information and browsing patterns are more encrypted and secure, thereby reducing the threat of a third party capturing their data over the wire (or over the air, as in most cases today). When users connect to the Internet by opening their browsers, they are connecting through an https secure site.

Operators of guest networks and hotspots have typically redirected this first request to their own captive portal, to require users to accept their terms and conditions, to charge for Internet access, or simply to authenticate them as a valid user. When the users' homepage is set to an http (non-secure) site, there is no issue. But if the redirection happens on an https (secure) site, there are some hurdles to overcome.

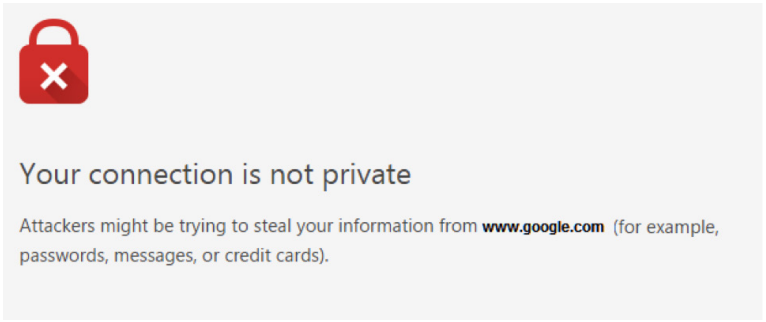
These hurdles arise due to certificate encryption and how the browser interacts with the content of the website to make a secure connection. Upon initiating an https session, the browser will validate the certificate against known information (the domain/certificate name, validity of the certificate and the certificate signing authority) before accepting the data stream for the website. So if the certificate isn't validated (i.e., if the information the browser has does not match the certificate sent back from the requested site), then the website will not open properly. As a result, the user will receive an error message, which will vary depending on the browser.

Different browsers handle the certificate mismatch in different ways. Some browsers allow for an exception to be made, in which case they would let the user know there is an error with the certificate, an indication that the site might not be as secure as they expected. The message will look something like this:

Internet Explorer:



Chrome:



As stated above, the biggest issue is with the certificates not matching the information the browser is using to validate. There are several options for getting around the above errors for the end user.

1. **Pass through all https (secure) traffic and wait for an http (unsecure) request to redirect to the captive portal.** A large majority of gateway providers chose this option in the past, which worked when https sites were limited to such companies as banks and the like. Now that some of the more common sites are using https security, it's possible that users might never go to an http site from which they could be redirected. Imagine a user who goes online to check their Yahoo email or Facebook, or conduct a Google search. Unless that person clicks on an http link in their search, they will not be redirected. This is problematic for a site wishing to have its guests authenticate, accept terms and conditions, or charge for access.

2. **Obtain actual valid certificates.** This option requires that a company have control over the certificate being issued (not just the company that owns the certificate). This is problematic because it would allow people to not only impersonate the site but also grant them access to any of their other secured systems that use the same certificates. Additionally, the logistical challenge of trying to obtain certificates that match every possible startup Web page that is using https is not very practical.

3. **Dynamically create fake "valid" certificates on the fly.** This is a very technically challenging option because of the need to encrypt and enter the correct information the browser is seeking. It is also very questionable from a security standpoint because it means that, in essence, nothing would be truly secure. Not only does the company leave itself open to "man in the middle" attacks, this option means that no certificate/site would be secure on this network.

4. **Block https sites until after terms acceptance/authentication.** With this option most browsers will return a blank page or an error stating it can't find the page requested. In this case, users are connected to a network but their homepage is not being displayed. Since no message is delivered to guide them, they have no idea what to do.

5. **Use a certificate that is valid for the portal and redirect there.** In this scenario, the redirect will happen, but the end user will receive a certificate error warning from their Web browser, asking if they want to continue with the certificate that does not match the site they wanted to visit. With the heightened awareness surrounding personal security, most users will close the session. If they do continue on, as some browsers currently allow them to do, they will get to the captive portal and be able to accept the terms or to authenticate.

There is another option at play coming from outside the network. Some devices (Apple, Android) use a captive portal detection process, which notifies the user that a captive portal is being utilized on the network to which they are connected, and will sometimes open a browser to a standard http (unsecure) site. This helps in that the browser's first attempt is to an http site defined by the device/OS manufacturer.

CONCLUSION

Upon considering the aforementioned options to handle the redirection of https pages to a captive portal for guest and hotspot networks, none of them are particularly viable. As technology moves forward, manufacturers, vendors and public access service operators will need to work together to develop a more comprehensive and seamless solution.

Until then, the goal should be to educate users with in-room tent cards, room-key holder instructions or television tutorials to explain that their connection requires them to go to a non-secure page (such as the property's website) to obtain access.