BLINN.COMmunications
December 2019
179 Caren Avenue • Worthington • Ohio 43085-2520 • 718.577.1054 (NYC) • 614.785.9359 (Ohio)

Strong Security Enhances Website Trust

Sites that don't make on-line sales, accept credit cards, or gather personal identification information from users need no encryption. That's not an unusual point of view, but it's wrong.

Two areas deserve attention: Configuration files, stored files, and directories not intended to be public constitute the first area; how users connect to the website is the second. I'll dive into the second part first.

Why HTTPS is Important

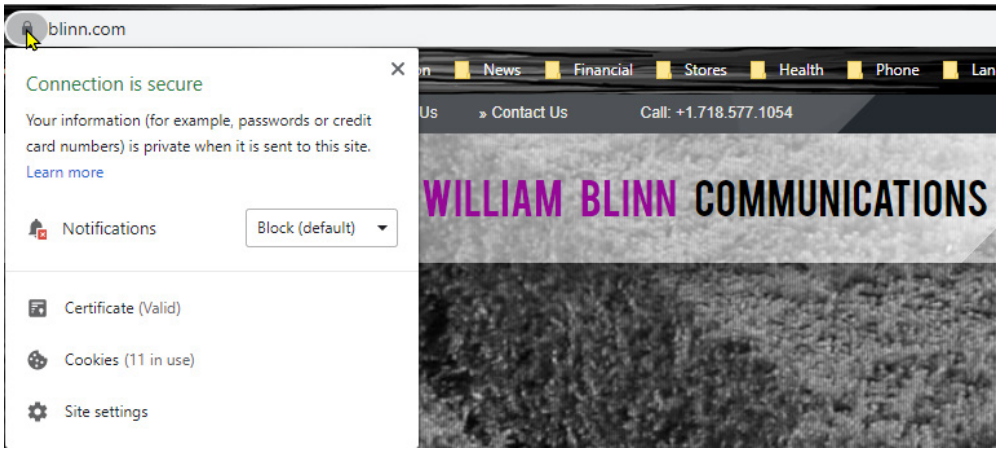
There are several reasons why sites should enable HTTPS, but the big one is this: Google reduces your site's visibility if it doesn't use HTTPS, so if you depend on Google to bring people to your site, you really need to enable HTTPS.

But there's more.

Using HTTPS keeps crooks from seeing information as it travels from a user to your website or from your site back to the user. An intruder could insert code in the data stream to trick a user into thinking that credentials are needed for site access, and then use social engineering to obtain the user's Facebook name and password.

Also, it isn't just sites that handle sensitive data that need to be protected. Snoops (such as internet service providers) can see traffic when it's sent in plain text and that might reveal information the user would prefer to keep secret.

Even for sites that collect or maintain no sensitive data, displaying a lock icon in the browser's address bar improves customer confidence. Those who visit will consider the site to be more trustworthy when the lock is present. One visitor is particularly important: The Google spider.



Viewing a site should display a lock icon at the left edge of the address bar. Clicking the lock should then indicate that the certificate is valid. All browsers have similar displays. This example uses Chrome.

How to Enable HTTPS

In the early days of the web, connecting to a website required typing the protocol (http://) and the full website address (www.blinn.com). Today all that's needed is the domain name (blinn.com) the browser and the website figure out the rest of it.

Well run sites will enforce HTTPS connections. HTTP means hypertext transfer protocol and HTTPS indicates that the connection is encrypted. If you type just a domain name (blinn.com), the connection starts in plain text and then one of two things will happen: Either a bit of code on the website will enforce an encrypted connection and your browser will show a lock icon at the left edge of the address bar or the connection will stay in plain text and the browser will display "not secure" or a lock with a red line through it.

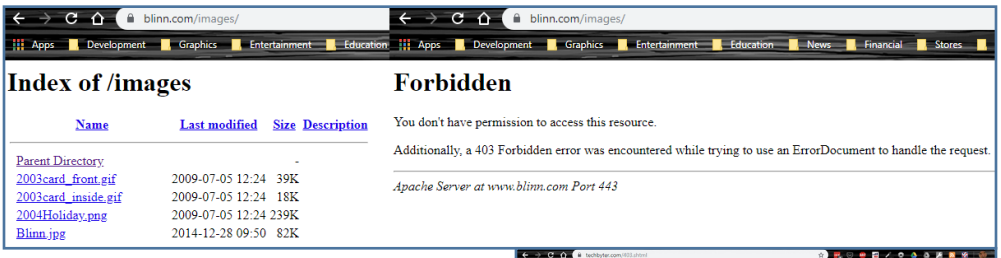
This is a relatively easy problem for website operators to correct. You'll need a security certificate, which is something that many hosting services provide automatically. If not, you'll have to purchase your own certificate from a certification authority (CA).

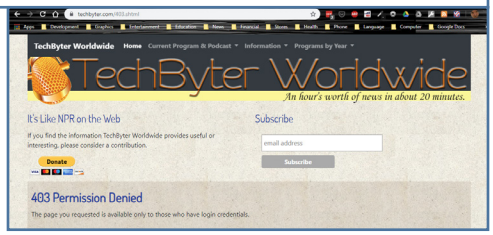
After installing the certificate, you'll also need to ensure that a special file in the site's root directory issues what's called a rewrite command to force a secure connection even if the visitor hasn't specified HTTPS.

You'll also need to ensure that every link on every page throughout the website refers either to local resources (script files, images, style sheets, and typefaces) or to external resources and websites using HTTPS instead of HTTP.

Not all external links accept secure connections and if your site contains an unencrypted link to an external resource, the browser will either display the lock with a red line or a warning that the site contains "mixed content".

Blinn Communications News (ISSN 2639-5762) is for clients and friends of William Blinn Communications, Worthington, Ohio. Phone 614.785.9359 (Columbus) and 718.577.1054 (NYC). On the Web at www.blinn.com.





Without appropriate protections, visitors to a website can list the contents of directories (left, above). Adding a command to a special file in the wesite's root directory will prohibit the display but the message is terse (right, above). Another option redirects the user to a "403/ Forbidden" page that can be styled to match other pages on the site (right).

The mixed content problem can be resolved by working with the external content provider to establish secure connections, by downloading the resource so that the connection is local, or (in extreme cases) eliminating the resource or the external site link.

Protecting Site Files

Most websites have directories that are not intended to be accessible directly.

Here's a trivial example: Your site probably has a directory where images used on the site are stored and you may not want casual users to open the directory with a browser and see a list of the files. It's easy enough for people to "borrow" your images, so there's no reason to make it easy for them.

Adding "Options -Indexes" to the site's .htaccess file will turn off directory listings but the message is curt. In some cases, a better option involves creating a file in the directory to redirect the user to an error page that matches your site's design.

When directory listings are allowed, site visitors can see more than just the site's images. If you have code files in an accessible directory, a visitor could display the contents of a script file and learn how to gain access to a database used by the site. Developers who use the Perl language usually place their files outside the site's root directory so that no browser will have access to it.

PHP, another commonly used development language takes a different approach. PHP files are accessible to browsers, but they don't directly return content to the browser. Instead, the PHP file communicates with the PHP application on the server, receives output from the PHP service, and returns that information through the web server to the visitor.

PHP files may contain login credentials and links to files that should not be exposed to site visitors. While this type of security is sufficient for many websites, it should not be used where full security is essential.

Depending on the Host

One of the easiest but most critical safeguards is making sure that anything installed on your web server is up to date.

Bluehost, the service most of my clients use, keeps the base software patched and updated. As with virtually all site hosting services, Bluehost runs open-source software that is frequently updated: The Apache web server and MySQL database on top of the Linux operating system.

Few hosting operations will update any applications that you have installed. If you have installed WordPress on your site, it's up to you to check for updates regularly and install them. The hosting companies won't do this because updates can break existing installations if those installations have been improperly modified by the user.

Some content management systems such as WordPress accept plug-ins that can improve security and these must be updated frequently, too. Some plug-ins can introduce security flaws of their own.

Those who operate websites that collect financial or personal identification information must take security very seriously, but even informational sites can't overlook it. Ω

Microsoft's Edge

Microsoft is dropping its proprietary display engine in the Edge browser and switching to Chromium.

It's not yet fully baked, but anyone who wants to have a say about how the new browser will work can do so by installing the version that's still being developed and providing feedback to the developers.

You have three choices: Beta, Dev, and Canary. The newly released beta channel has the most stable version and is updated at six-month intervals.

- If you want to take a look and still avoid the most serious bugs, you'll want the **Beta channel**.
- The **Dev build** is the right choice for those who want to see the newest work and won't be bothered by more numerous bugs. Updates are issued weekly.
- For the more adventurous, there's the **Canary build**, which is updated daily and is therefore the least stable.

Download any of the three versions from Microsoft's website. The Edge Insider program is available for users of Windows 7, 8, 8.1, and 10, and also for the MacOS.

Those who have used Chrome will see a lot of familiar icons and menus. One difference, though, is that plug-ins and extensions for Edge are served from the the Microsoft Store. Also, unlike the Chrome browser, only 130 or so extensions are listed for Edge. This will increase over time.

Every website page I've examined with the Chromium-powered Edge browser displays properly. Microsoft hasn't announced plans for when the new browser will replace the old browser in Windows 10 other than to say the switch will be made when the new Edge browser is stable enough for production use.

If you want to take a look, check it out at *microsoftedgeinsider.com*. Ω