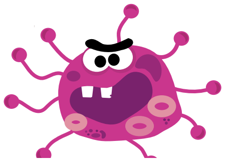
How to be a responsible website owner.

A few simple things you should do to protect your users from being hacked.

Cybercrime is rampant (unfortunately an understatement.) There are millions of bots invading every corner of the internet looking for information about us that can be sent back to their masters, who then will use it to harm us. I wish I could be saying I am exhibiting paranoia, but pick up the paper and you will see reports of hackers every day. (Almost all hacks go unreported for various reasons, so the actual number is enormous.)

If your website contains any information about your clients, or it allows visitors to set up a user account, you should do everything you can to protect this data. There are two threats to the data:

The Bots



a) They wander the web scanning all the data they come across.
b) If they find sites which allow the creation of user accounts, they try setting up one for themselves so they are partially inside your system, hoping to find other doorways ajar.

The Users Themselves

Despite all the articles in the news, many users still have extremely bad habits. The worst of all is using the same username and password on all their sites. This means if one of the sites is hacked, the bots take their payload of usernames/passwords and try to get into other sites hoping to gain access to government/business/family secrets, money accounts, or to encrypt files to gain ransom money.

How to Stop the Bots

SSL Certificates [Secure Socket Layer]

These are certificates you can buy from the company which is hosting your site. The cost is between $0 and $100 per year depending on the company, your plan, etc. Your hosting company will add this to your site. Thereafter, your site, e.g., mysite.com, will be;

https://mysite.com instead of ***http://mysite.com***

Browsers automatically add the proper prefix when a user enters: 'mysite.com'.

Http and *https* are protocols for sending messages between computers. HTTP just sends what ever text you type in. Anyone can (and does) read your messages.

HTTPS is 'secure' because it encrypts the message so anyone intercepting the message cannot read it. This helps keep your information safe from simple easedroping or from bots grazing on the web.

Side note:

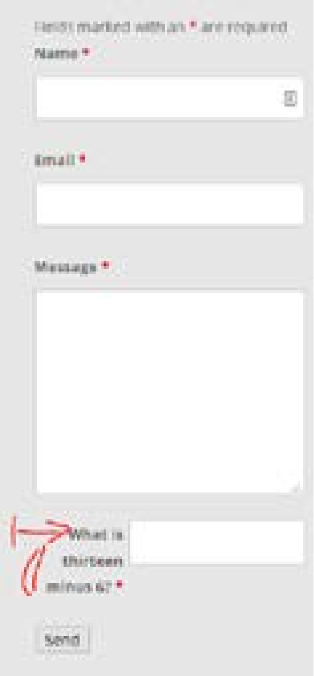
> *E-mail is not encrypted and is often 'owned' by the company storing it for you. So you should not expect privacy for anything you write in an e-mail.*

Use CAPTCHA

Completely Automated Public
Turing test to tell Computers and Humans Apart

CAPTCHA are those little irksome tests that come up when your are trying to perform a task on a site. You want to have your Web Designer to put them on pages where you are asking the visitor for information. In particular, make them do one before you accept their input for the creation of a user account.

On my first site, I let anyone create a User Account. After the first week I had over 400 users. All of them (Russian) Bots.



Fields marked with an * are required

Name *

Email *

Message *

What is thirteen minus 6? *

Secure Your Client's Passwords

If you allow clients to create user accounts on your website, you need to protect their passwords.

You can test if your website is doing this already. Create a user account for yourself on your site. When logging in, indicate that you have lost your password.

If your site asks you to reset your password, you are [probably] good.

If your site sends you a reminder of what your current password is, your site is broken. You must immediately have your web designer fix this. You may have to move to a different web builder program.

This is Important. Do This!

Click for more information about resetting passwords:
https://www.troyhunt.com/everything-you-ever-wanted-to-know/

Optional Technical Stuff

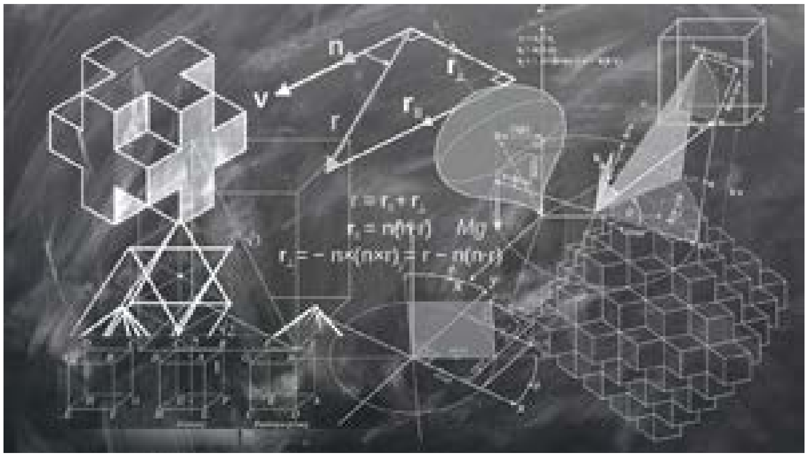
How Certifcates (are supposed to) Work.

A SSL Certificate is a certificate which says you are who you claim you are. You get one from a trusted Certificate Authority who verifies your identity before isssuing the certificate. The certificate is contained in a small data file which binds a cryptographic key (just a unique long string of funny numbers) to the name of the person in the file. You can't change the file without voiding its contents.

Everything is handle automatically by your browser. If you connect to a site that has a certificate, your browser and the site pass back and forth (handshaking) a bunch of data which allows them to generate a shared key which will be used to encode messages sent between them. Only your browser and the website have the new key, so only the two of them can read the messages. If anyone intercepts the messages, they just look like nonesense messages.

For more detail search for 'SSL certificate'.

Note that possession of a certificate only verifies the claimed identify of the possessor and not anything about other attributes of the possessor. Anakin Skywalker's certificate only says that the issuing company verified that the person they sent the file to someone called Anakin. He may be on the side of light or darkness. You will have to find out from your interactions (Alert: Don't order the Death Star!). Any messages sent via this certificate will only be readable by the person to whom the certificate was issued.



How Sites Safeguard Passwords

A site protects passwords by throwing them away and never storing them. That way, the passwords can never be stolen. But how can this be because the site must ‘remember’ the password to let you in the next time you login.

How can they know it and not know it? It is all done by the Mathemagics.

One-Way Functions

These mathematical functions are also called Hash Functions because they take a text string, for example, your password, and turn it into a gigantic funny number.

Password --> =] Hash function[= --> big, big funny number (***BBFN***)

This BBFN has some funny properties.

Even if you completely understand how the Hash Function generated the BBFN, you cannot get back from the BBFN to the original text. It is a one-way trip, [You can get it back, but the expected time to find the original password is the life-time of the Universe, so we say it is one way.]

Another property of the function is that for two text strings that are different in any way, not matter how minor, completely unrelated BBFNs are generated. The function will never produce the same BBFN for two different text strings. Each BBFN is unique. (Once again not actually guaranteed to be unique, but if you hashed text strings from now until the end of the universe, with luck, you might see one duplication.)

Storing BBFN

So you create an account and enter your password. The site hashes your password, stores its BBFN, and throws away the password.

Later, you log into the site. You enter your password. The site hashes your entered password. If its BBFN matches the stored BBFN, then you entered the correct password.

Even later, a hacker breaks in and steals the database from the site. The hacker now has all the usernames stored in the site, but no passwords, only BBFNs. The passwords are safe because she cannot deduce the original passwords from their stored BBFNs.