AddThis®

15 Strategies to Improve the Security of Your Small Business Website

A Beginner's Guide by AddThis®

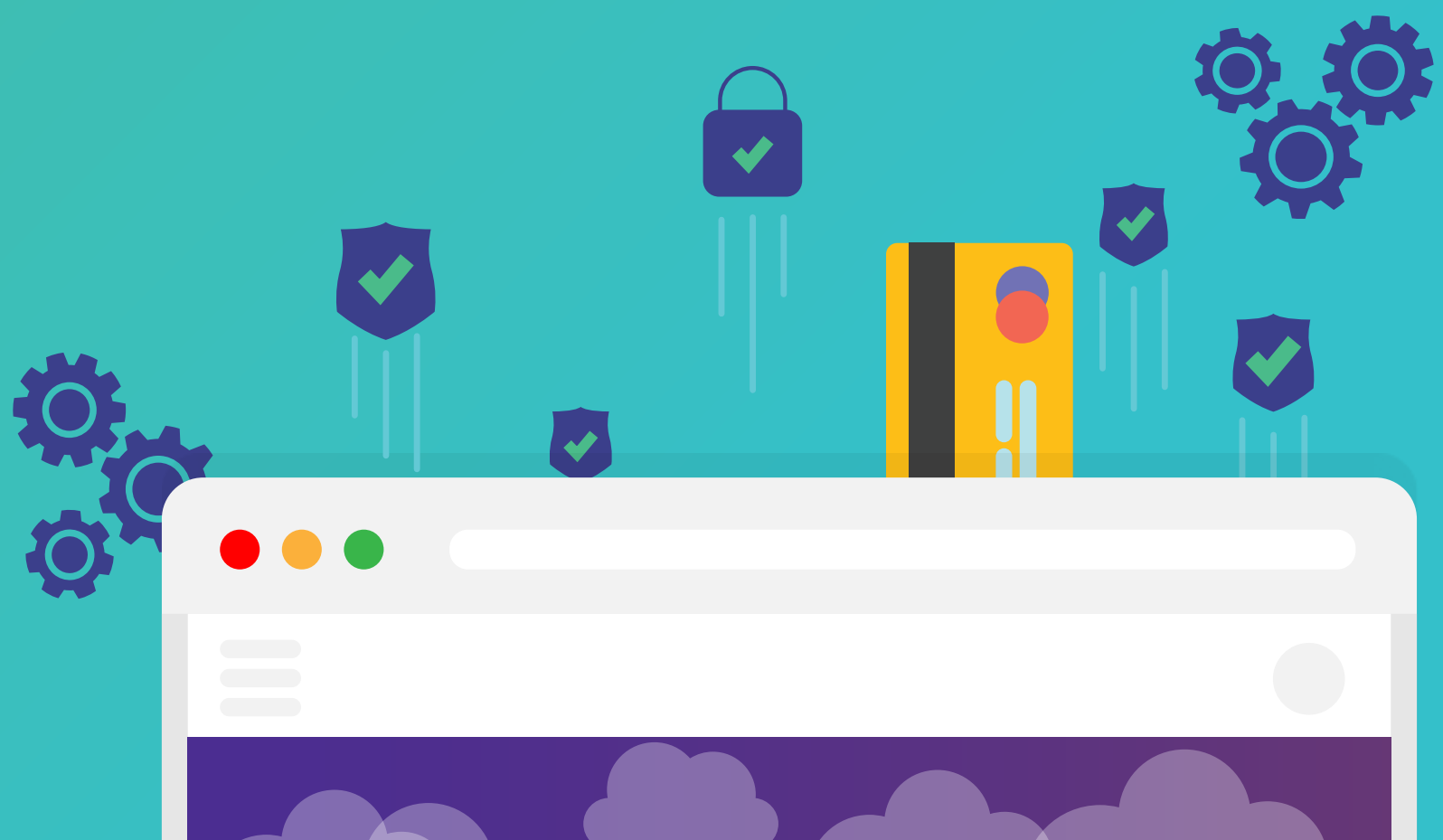


Table of Contents

So, You Want o Secure Your Business Website?

Introduction

It's no surprise that hacked websites and data breaches are a widespread and damaging problem in today's online world. Because websites are directly accessible 24/7 from anywhere in the world, they present an attractive target for cyber criminals. Sometimes hackers attempt to steal personal customer data, misuse credit cards, or perpetrate identify theft. Other times, they simply want to deface a site by hacking into it or taking it down with a Distributed Denial of Service (DDoS) attack.

To keep your site safe, it's critical that you or your website administrator invest the time and effort necessary to ensure your website's security is as robust as possible. This is especially true if you conduct business on the site—because a hack will almost certainly mean lost revenues—or if your site collects or stores personal customer information. A data breach can damage your reputation, scare away customers, and even get you into legal trouble.

If you can afford to hire a pro or enlist a consultant to focus on assessing, securing, and monitoring your site's cybersecurity, then your best bet is to skip to the last strategy in this eBook: hire a professional. We realize this might be impractical for many of our small business readers so we've packed this eBook with 15 strategies that every small business manager can use to strengthen their website's security posture.

1 Update Software

Security flaws in software are a fact of life. Hackers constantly try to discover "holes" in the security of commonly used platforms to steal data or harm systems. "White hat" hackers are people who try to do the same thing, but with the intention of helping software makers close the holes before they cause actual harm. Reliable developers pay close attention to all information that might indicate a security flaw in their software and then quickly "patch" the software to prevent harm to their users.

For this reason, it is extremely important to keep updating all the software involved in running your website. This includes the server's operating system, content management platform (such as WordPress or Joomla), all installed themes/plug-ins/extensions, commerce/store platform, forum systems, anti virus/malware tools, and so forth.

By using the most up-to-date version of each system, you're protected from the security flaws inevitably discovered over time. Many websites are hacked simply because they use older software versions.

Most serious software vendors offer ways to make it easier to keep track of available updates, whether by automatic software updates, notices in the admin console, a mailing list, or an RSS feed. It's worth plugging into at least one of these channels to know when there is an update to install.

Tip: Are you using a managed hosting provider?

If your site is running at a managed hosting provider, then it is their job to keep your software updated so that you don't have to worry about it yourself. Touch base with them occasionally to make sure they are on top of it, and find out if there might be portions of your website that they are not maintaining for you. If so, be sure to update those components yourself.

2 Backup Daily

The oldest—and still the best—strategy to protect yourself from nearly any computer-related disaster is to frequently backup all important data and systems. In worst-case scenarios such as a hacker infiltrating your system, malware destroying your data, or ransomware making your data useless, a recent backup will allow you to quickly return your website to its state just before the attack.

Minimally, ensure that your entire system—including all software configurations, website content, and database content—is backed up every night. In the event of an immediately evident disaster, you can revert to the backup and not lose more than 24 hours of changes to content or data.

However, because some problems only become evident after some time passes, it's important to retain a few weeks of daily backups. For example, if you discover that malware infected your system two weeks earlier, you can go back to the last "clean" backup and restart your systems from there.

How to perform backups depends on the particular systems you use. Your hosting provider might provide a backup service for some or all of your website and data, possibly at additional expense. Some platforms include built-in, regularly scheduled backup functionality for their own data. If they don't, there are often plug-ins or extensions available for popular platforms. There also are third-party backup services available. Finally, it makes sense to manually set up your own backup procedures in certain situations. Do some research to determine the best way to back up each part of your website. Simultaneously, don't forget to research (and record) how you can restore your systems from the backups, if necessary.

One guideline for backups is the 3-2-1 rule: Three different backups, on two different media (hard drive/cloud, hard drive/DVD, etc.), and at least one offsite.

A final piece of backup advice: Never store your backups on the web server itself or even at the same physical location as the server. The backups could contain unencrypted data or unpatched software that might aid a hacker in compromising your live website. Also, if a catastrophe (e.g., a fire) strikes at the site, make sure your backups are not lost together with the web server itself.

Tip: Backups help protect against other disasters as well

While this eBook focuses on malicious attacks against your website, other all-too-common dangers are incidents like hardware failure, accidental deletion of files, and disastrous configuration changes. Having a recent backup will make it possible to get your site up and running fast—and prevent unnecessary additional loss of revenues or reputation.

3 Enforce Strong Passwords

Requiring strong passwords is a sound security measure that, while relatively easy, is often overlooked. Companies often fail to change or remove the standard administrator username. Because today's hackers often use advanced password-cracking software, it is critical that every user account with access to website systems be impervious to **brute-force** and **dictionary-based**, password-cracking attempts.

Strong passwords combine three elements: they are long, complex, and unique. Specifically, we recommend enforcing a password policy containing these rules:

- Passwords must be at least 12 characters in length.
- Passwords must contain at least two alphabetical characters.
- Passwords must contain both lowercase and uppercase letters.
- Passwords must contain at least two numerical digits.
- Passwords must contain at least two special characters (such as & ^ % * $).
- Passwords may not contain any words in the dictionary or any commonly used IT login names (e.g., admin, password, administrator, sysadmin).
- Passwords may not contain any personal information (such as name or birthdate).
- Passwords may not be reused for multiple accounts.
- Passwords must be changed periodically (e.g., every 90 days).
- Passwords must not be stored anywhere on a computer or other electronic device in plaintext form.

Enforce strong password policies by:

- Defining the policy and deadline for implementation.
- Informing all relevant users of the policy and the deadline for implementation.
- Auditing all passwords after the deadline to find those out of policy.
- Generating random policy-compliant passwords for all nonconforming passwords and disseminating them to relevant users via a secure means.

Because just one weak password among your team can make your entire website vulnerable, *your password policy must be respected by everyone*.

Tip: Use a password manager

The biggest challenge with using long, complex, and unique passwords is, of course, how to remember them all. Actually, you and your users don't need to remember passwords at all. Instead, use a password manager, such as **LastPass**, **Zoho Vault**, and **KeePass**, to store and use all your passwords in encrypted form.

4 Implement Two-Factor Authentication

Most systems are still accessed by entering a single set of credentials, namely username and password. Two-factor authentication takes this one important step further by restricting access until two forms of identification are provided to the system.

Implementing two-factor authentication typically involves requiring every user to supply a combination of "what you know" and "what you have" before accessing a system. "What you know" is usually a standard username and password pair. "What you have" can be implemented in a number of ways, including **(1)** access to a mobile device or email account where a single-use passcode is sent for each login, **(2)** an electronic fob or smartphone app displaying a frequently changing code validated during login, and **(3)** a simple code card containing codes to look up during login.

While two-factor authentication adds a bit of inconvenience to authorized users, a major weakness of relying on a username/password combination alone is eliminated. This authentication makes it far more difficult for a hacker to discover credentials and then gain access to a server, website, database, or online tool.

Hackers use various techniques to obtain privileged user credentials, such as brute-force, password-guessing, and "social engineering" (see strategy 13). However, when two-factor authentication is implemented, having only one set of user credentials is worthless to the hacker. This is because to succeed, the hacker needs the credentials to the authorized user's smartphone, email account, or code card, a much more unlikely scenario.

Tip: No two-factor authentication support?

If your system does not natively support two-factor authentication, install a plug-in or extension that provides this functionality for your platform or consider changing over to a platform with this feature.

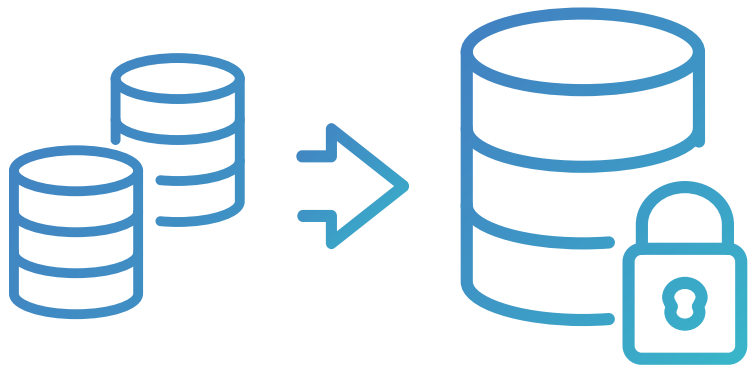
5 Use HTTPS

Unlike plain HTTP (the "HyperText Transport Protocol" used to communicate between web servers and browsers), HTTPS is a more secure way of exchanging data between web servers and end-user browsers. The "S" in HTTPS stands for "secure" and means the website's data flows are encrypted in both directions. Look closely at the URL of any web page and you'll discover if that page is delivered encrypted (https://) or not (http://).

The use of HTTPS assures users that **(1)** the website is really coming from the server they think they're accessing; **(2)** login credentials, form submissions, and other personal information cannot be captured by a third party while in transit; and **(3)** no one can view or alter the information flowing between the website and the browser.

There is another good reason to use HTTPS. As an incentive to encourage more website owners to switch to the more secure HTTPS, **Google announced** that Chrome soon will show a "Not secure" warning for all websites not using HTTPS. Beyond the clear security benefits, hiding this worrying notice from your visitors is another good reason to use HTTPS.

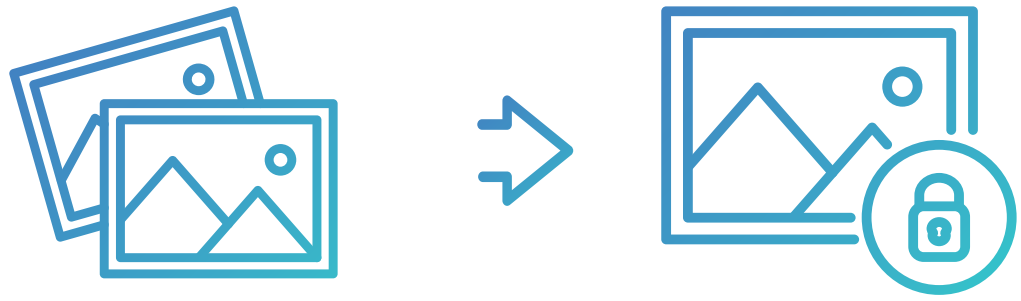
Moving to HTTPS is fairly simple for many websites. The first step is adding an SSL security certificate to your site, which you can do inexpensively by yourself or with your hosting provider. Some hosting companies will provide an SSL certificate for free or support free SSL certificates from **Let's Encrypt**.



The second step is to make sure all content loaded into web pages, including images and scripts, is shown using HTTPS. If even one element in a page is loaded with plain HTTP, the page is not considered secure by the browser.

Other steps you may have to take are beyond the scope of this eBook. If your site is more complex, find a good online guide (like **this one**) or let a professional handle this for you.

Please note: If you use the AddThis share counters, moving to HTTPS from HTTP will reset your website's share count. This is one consequence of installing a SSL security certificate on your site. Learn more **here**.



Tip: Using HTTP offers SEO benefits

In 2014, **Google announced** that sites using HTTPS will be given a slight advantage in how they rank in search listings. This means that, beyond the clear security benefits, switching to HTTPS also might help your site get more traffic.

6 Create Custom Admin URLs

Hackers know that most administrators never change the URL of their systems' admin login page, which makes it easier to crack the login page. For example, the default admin URL for WordPress is https://yourdomain/wp-admin, for Joomla it is https://yourdomain/administrator, for Magento it is https://yourdomain/magento/admin, and for OpenCart it is https://yourdomain/admin.

By changing your admin path to something less obvious, hackers will have to work much harder to find your admin page before they can start attacking it. It's really like adding an additional password layer to your systems. Be sure to choose wisely. There are actually lists of commonly used alternative login URLs available for hackers to try!

Consult the documentation for each of your systems to learn how to make this change. It's sometimes as easy as changing a single setting in your existing admin interface, although you may need to install a plug-in or manually change configuration files.

Make sure your custom admin URL is not discoverable. For this strategy to be successful, you must ensure that anonymous website visitors cannot discover your custom admin URL. This means not including it in publicly accessible files, such as robots.txt or log files, and that no failed-login processes redirect to it.

Tip: Don't forget your custom admin URL

Make careful note of the new admin login page URL you choose so that you don't end up locking yourself out of your own site's admin!

7 Limit Login Attempts

The two most common ways attackers attempt to hack into a system are by using automated brute-force or dictionary-based attacks to log in. This means they use software designed to try logging in with millions of combinations of usernames and passwords, whether by trying random combinations of characters or known words and phrases. These attacks can be remarkably effective, sometimes successfully breaking into a site within minutes or hours (especially if the password is not long, complex, and unique, as discussed in strategy 3).

An easy way to completely block this type of attack is to limit the number of failed login attempts allowed from the same client and/or IP address/range, within a certain amount of time. For example, if an incorrect password is entered five times within two minutes, then that client will be unable to try logging in again for 15 minutes. While unlikely to ever affect an authorized user, this simple measure can significantly reduce the chances of a successful brute-force or dictionary-based attack. The software can only try five combinations every 15 minutes, usually far too few to ever end up guessing the correct login credentials.

This is not foolproof, however, as sophisticated hackers will distribute their attack among many clients at disparate IP ranges, allowing them to circumvent this mechanism. Yet, even with a distributed attack, their efforts will be significantly hindered by the small number of login attempts permitted during any given period of time.

In addition, if you use third-party services like MailChimp or AddThis, do not share accounts with other members of your organization. Instead, have these individuals create an account to share access to the profiles.

Learn more here:
http://www.addthis.com/academy/what-are-addthis-profiles/

Tip: No support for limiting login attempts?

If your platform does not natively support limiting login attempts, search online for plug-ins or extensions that provide this functionality for your system.

8 Avoid Shared Hosting Environments

For budget-conscious small businesses, it's tempting to use a shared hosting plan. After all, they offer everything you need to host a website (and email and more) for just a few dollars a month.

The problem is that shared hosting is inherently insecure, because your website is running on the same server together with dozens or, more likely, hundreds of other websites. A hacker need only infiltrate a single unsecured site on the server to potentially gain access to (or infect with a virus) all the sites on that same computer. Even after the hosting provider cleans up the mess after a breach or infection, the passwords on all of the accounts must be changed to prevent easy reinfection, which is a big problem in many cases.

Likewise, a buggy or malicious program running on any of the websites can gobble up all available memory, disk space, CPU power, or internet bandwidth to slow your site to a crawl or freeze it entirely. Another common problem is when one site on a server sends out spam. All the websites on that same server (i.e., same IP address) can get blacklisted by anti-spam systems.

The bottom line is the "attack surface" of shared hosting servers is far wider than that of dedicated servers (or even virtual private servers) and so are the risks. We recommend avoiding shared hosting environments despite the higher costs of more secure environments.

Tip: Choose a reputable hosting company

Doing some research before selecting a secure and reliable hosting provider is super important. Not all hosting providers are as concerned about security as others. Regardless of which type of hosting you choose, make sure that the company hosting your website is serious and proactive about protecting their customers from cyber threats.

9 Monitor File Changes

Once a hacker gains access to your website, chances are this individual will make changes to files—inserting links to other sites, stealing data, vandalizing, installing malware, modifying existing files, adding new files, or deleting files. By constantly monitoring changes to files on the web server, you will have an early-warning system for possible break-ins to your site.

Changes made by developers, content authors, and content editors are, of course, fine. When modifications not made by your team are detected, it is a likely sign your site was compromised.

Installing a file change monitoring tool, also known as a File Integrity Monitoring (FIM) tool, is a good idea. There are lots of FIMs available for every operating system on a website platform. Some are stand-alone FIMs, while others are included as part of security plug-ins that also contain other features. These tools all provide a log—so you can review and audit all file changes made during any period of time—and most offer real-time alerting based on criteria you can customize.

The biggest challenge with monitoring file changes is filtering out the legitimate changes so you become aware of any suspicious changes. Otherwise, you will quickly ignore the reports or notifications your FIM sends you. Each tool has its own ways of doing this, and each server and website platform has its own file structure. It's worth spending the time to find the ideal balance between monitoring too many files (and getting insignificant alerts) and too few (in which case you might miss something important).

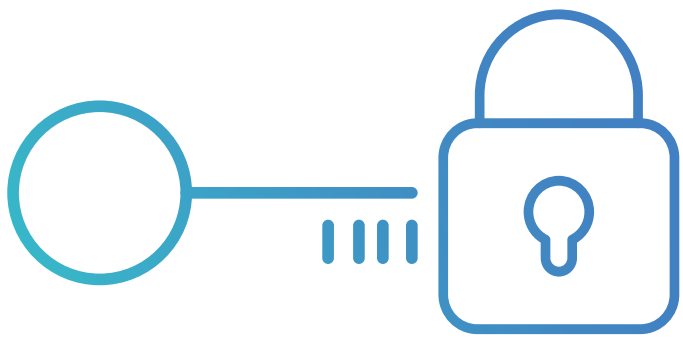
Tip: Prepare a recovery plan in advance

In the event that your FIM discovers an actual breach of your website, you will need to quickly revert the site back to the most recent uncompromised backup and change all of your passwords. To do this quickly, prepare a detailed, step-by-step plan in advance. This will make it faster, easier, and less stressful to handle the worst-case scenario of an actual hack.

10 Install Security Plug-Ins

Security plug-ins exist for most popular website platforms. These are modules that actively try to prevent hacking attempts in a number of ways, such as virus/malware scanning, blocking login attempts from known hacker IP addresses, blocking all logins during non-working hours, alerting on file changes and other topics covered elsewhere in this eBook (limiting login attempts, enforcing strong passwords, two-factor authentication, hiding the admin login page, making off-site backups, and monitoring your web server).

You think if these plug-ins do so much, why wouldn't I use them? In many cases, you should. However, as with any "jack-of-all-trades," you may find products that provide better individual solutions. Also, you could do many of the things offered by these paid solutions for free, whether in your platform's own admin console or by modifying some files. Plus, your site may have components running that are incompatible with these plug-ins. Finally, these plug-ins focus mostly on login security, but they don't address most types of issues involved in exploiting weaknesses in themes, scripts, other plug-ins, and other software running on the web server.



Tip: Beware of a false sense of security

While most small business websites will benefit from having a security plug-in, don't let having one give you a false sense of security. It is still important to consider all of the security aspects described in this eBook.

11 Deploy a Web Application Firewall

A web application firewall (WAF) sits between your website and the internet to inspect all incoming traffic and block anything deemed suspicious before it ever reaches your site. A WAF works by detecting common attack patterns, including widespread hacking methods such as **SQL injection**, **cross-site scripting**, **malware/code injection**, and **vulnerability exploits**. A properly configured and managed WAF will protect your website against these kinds of attacks.

Just a few years ago, all firewalls were physical (hardware) devices installed on a network. Hardware, or network, WAFs are still an effective option, though they are more expensive than the newer generation of software WAFs and require more expertise to deploy and manage. Conversely, they tend to have higher performance, which is important for large-volume websites that might otherwise slow down because of the time it takes to analyze incoming traffic before allowing it to reach the website.

Software WAFs are less expensive and more customizable. They are usually installed on the web server itself or as a plug-in to a website platform. Many web server hosting companies offer their own WAFs, in either managed or DIY configurations.

An additional option is to deploy a cloud-based WAF to direct all your website traffic (by changing your website's DNS configuration). Cloud-based WAFs may be a good choice for businesses who prefer a subscription-based, fully managed service.

Unless you (or someone on your team) are particularly technical, consider asking your hosting provider or a consultant to research and implement a WAF for your website.

Tip: A web application firewall is not a panacea

While having a WAF is important for your website's security, it is just one additional layer in your website's security posture. Deploying a WAF does not diminish the importance of any other strategy discussed in this eBook.

12 Manage Your Users

Let's leave technology behind for a bit and talk about the human factor. Because so many successful hacking attempts are traced back to someone in the organization, pay attention to the people with access to your website's administration. This includes all developers, content people, and administrators, whether in-house employees or outside vendors. Of course, if you are the only person with access to your website's admin console, then this topic is less important for you.

Least privilege. This strategy ensures that only the people who really need administrative—or "privileged"—access to your systems have it. Assign the minimum permissions level required by each person to do their job. For example, blog authors should only have authoring privileges in your site (and perhaps permission to edit their own posts), but not admin-level permissions. This is a key way to minimize the attack surface of your site.

Permissions. When someone needs admin-level permissions for a particular task, revoke those privileges when they are no longer needed. Far too many websites have long lists of defined admin users that do not require access (more about this in strategy 14).

Account sharing. Do not allow users to share accounts. Sharing accounts makes it impossible to determine, if something happens, who is responsible or at least whose account was used. Each user should have his or her own login and password. Assign individual account names and not generic login names such as "admin1" or "blogauthor3."

Lastly, keep an eye on user activity by reviewing your site's admin logs. If you see behavior that looks strange, such as logins during non-work hours or changes made to old files, it could be a red flag deserving further investigation.

Tip: Insider accounts are your greatest security threat

Security incidents involving malicious and inadvertent insiders are actually **more common** than breaches by outside hackers.

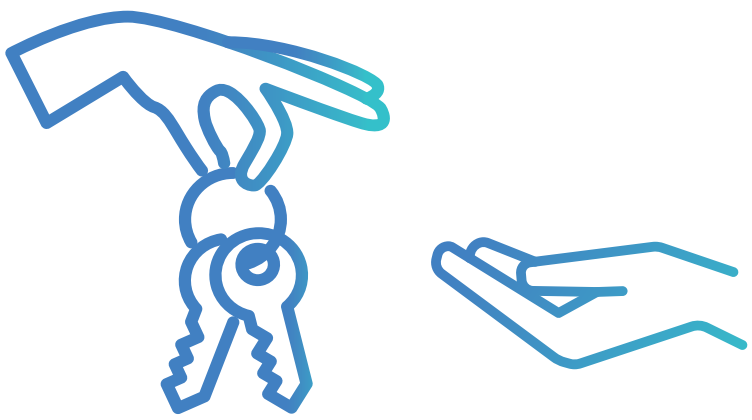
13 Recognize and Resist Social Engineering Attempts

Social engineering refers to the psychological manipulation of a person into divulging confidential information to an unauthorized third party or performing activities that do so. Due to its high degree of success, hackers frequently use social engineering techniques to obtain network login account credentials from employees at a targeted business.

Here are two examples:

- A hacker sends an email to an employee making the email appear to come from the employee's boss. The email urgently requests, for some purported purpose, the employee's username and password for logging in to the company's website admin console.

- An email requests the user click a link and then log in to an admin console. The link points to a counterfeit web page designed to look just like the company's real admin console page, from which the hacker can grab the credentials typed in by the unsuspecting user. This is called phishing.



Once the hacker receives credentials, he or she can access everything that the user account's privilege level provides. If the account has administrator-level permissions, the hacker can lock other users out of the site, vandalize the site, steal customer data, and more.

Because social engineering is so effective among unsuspecting employees, it is imperative to train your employees (and relevant consultants) to recognize and resist these kinds of social engineering attempts. Teach employees to never provide their login credentials to anyone, under any circumstances, even if their boss's wife calls on the phone in the middle of the night declaring a major company emergency. Train employees to never click links in emails before verifying the sender and legitimacy of the link. Employees must know to never send their credentials by email or to type them into any web page they reached via a link in an email, SMS, Facebook message, etc. Finally, inform employees whom to contact when they suspect a social engineering attempt was made.

Tip: Also beware of "offline" social engineering attempts

In more aggressive social engineering approaches, the perpetrator may try to gain the trust or cooperation of an employee using threats or bribes. In these situations, the perpetrator may attempt to convince the insider to provide credentials, copy/transmit sensitive data, or cause other harm while reassuring them that no one will ever know or they are not even committing a crime.

14 Define and Implement Exit Policies

Experience shows that whenever an employee will soon leave the company—especially if the person's employment was terminated by the company and/or the employee is disgruntled for any reason—the risk of data theft or system vandalism increases significantly. The same applies to outside consultants whose engagement with the company is terminated.

While these types of "insider threats" will hopefully never affect your business, it is prudent in these situations to take some of the following precautions to protect your website—and all of your IT assets.

- Remind departing employees and consultants of their confidentiality obligations.
- Increase monitoring of their actions when logged into sensitive systems.
- Block access to sensitive data and systems unless absolutely necessary for the duration of their employment. This includes both internal login accounts and any accounts the user might have to cloud-hosted applications or databases.
- Ensure that all login accounts (e.g., network, server, website, email, cloud application/storage) are closed when the person's employment ends.

Do

Tip: Don't forget about "soft" accounts

Beyond access to your company's internal systems, remember also to revoke access for departing employees and consultants to other accounts to which they might have the right to use, such as company social media channels, file sharing accounts, and teleconferencing systems.

15 Hire a Professional

Whether or not you address some or all of the strategies presented in this eBook, consider retaining the services of a professional security consultant or firm, or, especially for larger businesses, even hiring an IT manager specializing in website security. While you can deal with many of the smaller issues yourself, there are more advanced security measures better handled by an expert.

Furthermore, the internet security landscape is constantly changing and evolving—as is your website. The threats against which you must defend your site today may be different tomorrow. The only way to truly keep your website as secure as possible is to continuously monitor both your site and the new threats that will, unfortunately, continue to appear.

There are many consultants who specialize in ongoing security monitoring, as well as periodically assessing the security of your website. They use techniques such as vulnerability scanning (running tests against the site to reveal known security holes that hackers might exploit) and penetration testing (more aggressive testing during which an expert actually attempts to breach the website as a hacker might). Ideally, the former should be done two to four times a year and the latter once a year.

The results of these periodic evaluations are used to determine which steps are required to fully secure your website. Remediation may include procedures described in this eBook, as well as possibly more far-reaching moves, such as relocating where your server is hosted, replacing existing software systems, or implementing different organizational policies.

It is crucial that you select a security specialist you trust, so take your time researching and evaluating your options. Your goal is to develop a close working relationship based on assurance and cooperation so your website is fully "hardened" against any attack that the "bad guys" might try to launch against it.

Tip: An expert can provide rapid remediation

Having a relationship with a trusted security expert or firm provides the added benefit of having someone intimately familiar with your systems to quickly diagnose and remediate the problem.

Conclusion

The potential damage to a business from a website security breach can range from annoying to devastating. All small business owners need to carefully weigh the possible consequences in their own situation and make the corresponding level of investment to ensure their website is as safe as it needs to be from the hacker threat.

We are committed to the safety and security of all the AddThis tools. Please check out our AddThis Academy to find more technical information or if you have any questions, contact our support team: **http://www.addthis.com/support**