Identify the vulnerability

As there may be multiple, independent hacks in place, even if you're able to find and fix one vulnerability, we recommend continuing to search for others. Start your investigation by reading about the top ways websites get hacked by spammers.

You'll need

- Shell/terminal administrator access to your site's servers – web, database, files
- Knowledge of shell/terminal commands
- Understanding of code (such as PHP or JavaScript)
- Ability to run two antivirus scanners

Next actions

We'll cover several common ways a site can be compromised. Hopefully, one of these vulnerabilities will either apply to your site or will shed light on additional possibilities.

Please be aware that **vulnerability scanners** differ from **antivirus scanners**. Vulnerability scanners can be far more invasive and have greater potential to cause unwanted damage to your site. Please follow all directions, such as backing up your site, before running the scanner.

Potential vulnerabilities

Potential vulnerabilities to investigate include:

Virus-infected administrator’s computer

On an administrator’s virus-infected computer, the hacker may have installed spyware to record the site admin’s keystrokes.

- **Check for viruses on administrator’s systems.** We recommend running several reputable antivirus scanners, or AV scanners, on every computer used by an administrator to log in to the site. Since new malware infections are constantly being designed to evade scanners, this action isn't a foolproof method of virus detection. Since AV scanners might report false positives, running several scanners can provide more data points to determining whether a vulnerability exists. Also consider scanning both your webserver and all devices used to update or post to the site, just to be safe.
    - If the AV scanner detects spyware, a virus, trojan horse, or any suspicious program, investigate the site’s server logs to check for activity by the administrator who owns the infected computer.
    - Log files may have been altered by the hacker. If not, correlating the administrator’s username with suspicious commands in the log file is further evidence that a virus on an administrator’s system caused the site to be vulnerable.

Weak or reused passwords

Cracking a weak password can be relatively easy for hackers, and it provides them direct access to your server. Strong passwords have a combination of letters and numbers, punctuation, no words or slang that might be found in a dictionary. Passwords should only be used for one application, not reused throughout the web. When passwords are reused, it only takes one security breach-on one application for a hacker to find the login-and-password then attempt to reuse it elsewhere.

- In the server log, check for undesirable activity, such as multiple login attempts for an administrator or an administrator making unexpected commands. Make note of when the suspicious activity occurred because understanding when the hack first took place helps determine what backups may still be clean.

Out-of-date software

Check that your server(s) have installed the latest version of the operating system, content management system, blogging platform, applications, plugins, etc.

- Research (perhaps through a web search) all installed software to determine if your version contains a security advisory. If so, the possibility that outdated software caused your site to be vulnerable is quite likely.
- As a best practice, always aim to keep your servers' software up to date, regardless of whether outdated software resulted in vulnerability issues this time.

4. Permissive coding practices, such as open redirects and SQL injections

Open redirects

Open redirects are coded with the intention for the URL structure to allow the addition of another URL so users can reach a useful file or webpage on the site. For example:

```
http://example.com/page.php?url=http://example.com/good-file.pdf
```

or

```
http://example.com/page.php?url=<malware-attack-site>
```

- If your site is abused by open redirects, you likely noticed the message in Search Console provided example URLs that included open redirects to an undesirable destination.
- To prevent open redirects in the future, check if "allow open redirects" is turned on by default in your software, whether your code can prohibit off-domain redirects, or if you can sign the redirect so that only those with properly hashed URLs and the cryptographic signature can be redirected.

SQL injections

SQL injections occur when a hacker is able to add rogue commands to user input fields executed by your database. SQL injections update records in your database with unwanted spam or malware content, or they dump valuable data to output for the hacker. If your site uses a database, and especially if you were infected with the malware type SQL injection, it's possible that your site was compromised by a SQL injection.

- Login to the database server and look for suspicious content in the database, such as otherwise regular text fields that now show iframes or scripts.
- For suspicious values, check that the user input is validated and properly escaped or perhaps strongly typed so they can't be executed as code. if user input isn't checked

before database processing, SQL injection may be a root-cause vulnerability on your site.

Previous
← Assess malware damage

Next
Clean and maintain your site →

Last updated December 22, 2017.