Decrypting AzoRult traffic for fun and profit

By NexusFuzzy

Published: 2021-02-06 · Archived: 2026-04-05 22:54:21 UTC

5 min read

Feb 6, 2021

Press enter or click to view image in full size

There will be times in your career when you will be presented with a traffic capture and get the task to determine what happened and if any data was stolen.

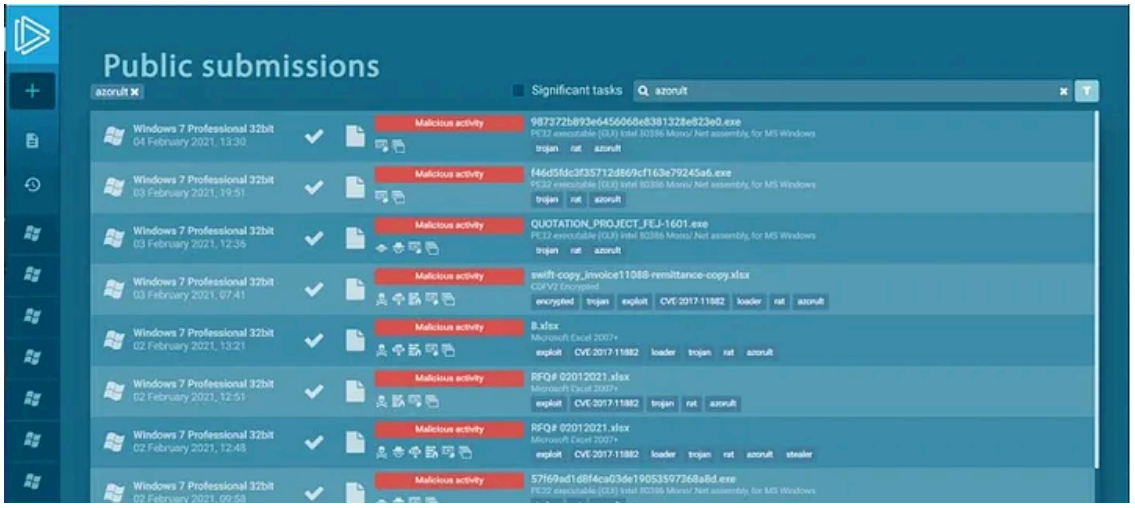
In this post, I will show you how you can squeeze all those juicy information from a PCAP traffic capture from an Azorult infection.

At the end, you will be able to answer which data has been stolen so you can act accordingly. Let’s start!

Getting sample data

Head over to https://any.run and search for “Azorult” in public submissions or use the PCAP you already got

Press enter or click to view image in full size



Most likely you will find a lot of samples

You will find a lot of samples without actual network traffic since the command and control server was already offline when any.run analyzed the sample. Have a look at samples which show POST requests

Press enter or click to view image in full size

You might need some patience

Now comes the fun part! As you might have noticed, the POST request data is encrypted in some way. Turns out, it is just XORed with a 3 byte key which unfortunately is not the same for all variants. What now? Make “some” educated guesses?

Get NexusFuzzy’s stories in your inbox

Join Medium for free to get updates from this writer.

Remember me for faster sign in

Fear not, I created a tool which first tries to decrypt it with keys I found in the wild and if this is not successful, it will start to brute force the key. This is possible with the help of a known plaintext attack since I learned through manually reversing AzoRult that the plaintext stolen data contains strings like “<info” which we can look for after every decryption try.

You can get it here: https://GitHub.com/hariomenkel/AzoBrute

Once downloaded, let it run against the extracted POST request and hopefully, you’ll receive the key.

Please consider creating an issue at the AzoBrute GitHub repository with your key so I can add it to the list of keys which are tried before trying brute force. Sharing is caring!

Once you have the key, copy it—you will need it for another tool

---