

Forum:[Hack Forums](#)

Thread: [How come the download speed is slower on Njrat in file manager??](#)

naxster - February 6, 2019, 11:29 am

Hello guys, i bought CrypticVPN 5 days ago, and it has been working great with the ports and everything.

But when i'm using Njrat, i notice that when i download file from File Manager, it's really slow! I mean can't barely get higher than 133kbit/s, and SOMETIMES it happens that the download speed is at 140kbit/s, but it goes right down to 133kbit/s right after.

Here is the picture for Njrat: <https://prnt.sc/mhkkwb>

But when i'm using Orcus RAT it downloads a lot faster!

Here is the picture for Orcus: <https://prnt.sc/mhkles>

Do you guys know why?

The only reason i still want to use Njrat, is because it's easier to crypt :)

In speedtest i get around 30mbit/s with CrypticVPN, so i don't think i have any speed issues since orcus rat downloads pretty good, or am i wrong?

? ?

Alles auf Rot - February 6, 2019, 11:46 am

It might be the problem of njrat socket and the way it handles data itself. only option is to modify the source, or switch the rat

Leo2Kirby - February 6, 2019, 12:48 pm

Disclaimer - I could be totally wrong about this, I only spent about 30 minutes working on it and would need more time to be sure

So I reviewed a public decompilation of njRAT and it looks like a few things might be causing it:

1. The server sets the maximum socket receiving size to 1MB, so each transfer is limited in size. I don't know how many transfers per second njRAT tries to make but it's still a relatively low buffer size regardless

2. Couldn't find the disassembly of the plugins it uses to perform the transfer (and not feeling up to tackling it with ILSpy right now), but I have read that .NET socket transfers can be very slow if you repeatedly read the file from the disk. If njq8 (the developer) decided to read the file in chunks (probably to decrease the client's memory footprint and time needed to start the transfer) then the disk could be bottlenecking the transfer, especially if the client doesn't have an SSD

3. It looks like (again, could be wrong here) njRAT encodes all its data as Base64 before doing the transfer, and Base64 generally makes things longer, so unless it's measuring the amount of Base64 data being transferred, you're actually transferring less raw data per transfer (if that makes any sense)

naxster - February 7, 2019, 5:33 am

(Feb 6, 2019 12:46 PM)Alles auf Rot Wrote: [\[CITE\]](#) It might be the problem of njrat socket and the way it handles data itself. only option is to modify the source, or switch the rat

The funny thing is that I've seen people download full speed when using Njrat :/

(Feb 6, 2019 01:48 PM)Leo2Kirby Wrote: [\[CITE\]](#) Disclaimer - I could be totally wrong about this, I only spent about 30 minutes working on it and would need more time to be sure

So I reviewed a public decompilation of njRAT and it looks like a few things might be causing it:

1. The server sets the maximum socket receiving size to 1MB, so each transfer is limited in size. I don't know how many transfers per second njRAT tries to make but it's still a relatively low buffer size regardless
2. Couldn't find the disassembly of the plugins it uses to perform the transfer (and not feeling up to tackling it with ILSpy right now), but I have read that .NET socket transfers can be very slow if you repeatedly read the file from the disk. If njq8 (the developer) decided to read the file in chunks (probably to decrease the client's memory footprint and time needed to start the transfer) then the disk could be bottlenecking the transfer, especially if the client doesn't have an SSD
3. It looks like (again, could be wrong here) njRAT encodes all its data as Base64 before doing the transfer, and Base64 generally makes things longer, so unless it's measuring the amount of Base64 data being transferred, you're actually transferring less raw data per transfer (if that makes any sense)

Wow that's great information! Thank you very much for that! :D

1. Do you think we can increase the buffer size?

Or is it even possible?

2. Hmm do you mean my hard drive (I have SSD) or do you mean the infected computer? :)

3. Everything made sense, so again thank you for all the research! :)

But the strange thing is that I've seen other people download with full speed :/

I wonder if there is an issue from the VPN side :/

I.M.P - February 7, 2019, 5:54 am

NJRat isnt as well coded as orcus. Its a matter of quality.

naxster - February 7, 2019, 7:00 am

(Feb 7, 2019 06:54 AM)Atroxcity Wrote: [\[CITE\]](#) NJRat isnt as well coded as orcus. Its a matter of quality.

Aww that's a real bummer :(

I always get 2/22 in Spyralscanner when scanning Orcus RAT. :(

It's always Avast and Kaspersky that's finding the virus! :P

Is that even good enough? Or is it really bad? :/

I.M.P - February 7, 2019, 7:00 am

(Feb 7, 2019 08:00 AM)naxster Wrote: [\[CITE\]](#) Aww that's a real bummer :(

I always get 2/22 in Spyralscanner when scanning Orcus RAT. :(

It's always Avast and Kaspersky that's finding the virus! :P

Is that even good enough? Or is it really bad? :/

thats pretty good. Let me Pm u!

Takehito Morokuzu - February 7, 2019, 7:43 am

(Feb 7, 2019 08:00 AM)naxster Wrote: [\[CITE\]](#) Aww that's a real bummer :(

I always get 2/22 in Spyralscanner when scanning Orcus RAT. :(

It's always Avast and Kaspersky that's finding the virus! :P

Is that even good enough? Or is it really bad? :/

When crypted? Or just after you built it

naxster - February 7, 2019, 8:04 am

(Feb 7, 2019 08:43 AM)-yuno gasai- Wrote: [\[CITE\]](#) When crypted? Or just after you built it

After it's Crypted, before crypting I get like 21/22 :/

Takehito Morokuzu - February 7, 2019, 8:53 am

(Feb 7, 2019 09:04 AM)naxster Wrote: [\[CITE\]](#) After it's Crypted, before crypting I get like 21/22 :/ Message the Crypter DEV.

Vuln - February 7, 2019, 9:06 am

Do you have the same problem when you have yourself as a client?

(Feb 6, 2019 01:48 PM)Leo2Kirby Wrote: [\[CITE\]](#) Disclaimer - I could be totally wrong about this, I only spent about 30 minutes working on it and would need more time to be sure

So I reviewed a public decompilation of njRAT and it looks like a few things might be causing it:

1. The server sets the maximum socket receiving size to 1MB, so each transfer is limited in size. I don't know how many transfers per second njRAT tries to make but it's still a relatively low buffer size regardless

2. Couldn't find the disassembly of the plugins it uses to perform the transfer (and not feeling up to tackling it with ILSpy right now), but I have read that .NET socket transfers can be very slow if you repeatedly read the file from the disk. If njq8 (the developer) decided to read the file in chunks (probably to decrease the client's memory footprint and time needed to start the transfer) then the disk could be bottlenecking the transfer, especially if the client doesn't have an SSD

3. It looks like (again, could be wrong here) njRAT encodes all its data as Base64 before doing the transfer, and Base64 generally makes things longer, so unless it's measuring the amount of Base64 data being transferred, you're actually transferring less raw data per transfer (if that makes any sense)

I doubt that any of that is the real bottleneck.

Leo2Kirby - February 7, 2019, 9:10 am

(Feb 7, 2019 06:33 AM)naxster Wrote: [\[CITE\]](#) The funny thing is that I've seen people download full speed when using NJrat :/

Wow that's great information! Thank you very much for that! :D

1. Do you think we can increase the buffer size?

Or is it even possible?

2. Hmm do you mean my hard drive (I have SSD) or do you mean the infected computer? :)

3. Everything made sense, so again thank you for all the research! :)

But the strange thing is that I've seen other people download with full speed :/

I wonder if there is an issue from the VPN side :/

1. Well the source code is out there so you totally could try modifying it if you want. You'd be looking for the ReceiveBufferSize property and you'd want to change it to something higher than 1048576 (1024^2). Though keep in mind if you modify the stub to run the output exe through ildasm (IL Disassembler) to generate the stub.il that njRAT needs to build the newly updated stub.

2. I mainly meant the disk on the infected computer, though now that you mention it depending on how njRAT was written (if njRAT doesn't try to receive more data until the previously received data is written to the disk), the controller's disk could also cause slowdown.

3. No problem

Leo2Kirby - February 9, 2019, 9:37 am

Hey - saw your PM, would reply but I don't have 25 posts yet and wanted to get back to you quickly (I hope it's not a problem that I'm not replying privately):

I was talking about modifying the source code, just search it on google and you'll find it (I'd link it but I need 50 posts for that!)

(just control-f and type "receivebuffersize" to jump to the code in question). Keep in mind if you're

going to modify it to also modify the line that defines numarray3 as a byte array to be the same size as the new receivebuffersize value. Also keep in mind that the higher you set these the more RAM you need per connection so start small and work your way up.

If I had to guess what's going on here is that since njRAT only writes 1MB at a time to your disk and it's received data memory buffer is also only 1MB in size, that something can't keep up with the 1MB/s transfer speed and is causing .NET to slow the transmission to let whatever it is catch up. If increasing the buffer's memory size fixes this then that's good, otherwise I'd have no idea what is causing it.

darkannubis - *February 16, 2019, 10:55 am*

some rat his fastest nanocore style or imminent even darkomet goes fast