





## Putting Together the RDPieces

**Brian Moran** 

Consultant

**BriMor Labs** 





## A Brief List of Topics

RDP - WTF?

(YOU: But Brian, we don't really see much of this right meow)

(ME: Perhaps, but this is why you should care)

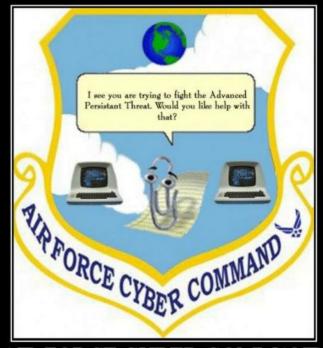
- Evidence
- Research
- Stuff with Things
- Profit?





#### I Feel So Seen

- Hello, my name is Brian Moran
- 13+ years Air Force career
  - 17ish years mobile exploitation & DFIR focus
  - Started BriMor Labs in 2014
    - Very happy since!



AIR FORCE CYBER COMMAND

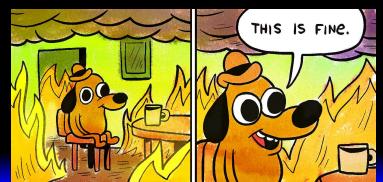
Yeah, we're screwed





## I Feel So Seen (Cont)

- You may know me from a variety of things, but I am very proud of the #DFIRFitin2020 challenge that was organized with the help of Kat Hedley (@4enzikat0r)
  - You can still join us! Details at https://www.dfirfitin2020.com
- Throughout 2020, our #DFIRFit4Good events have raised over \$10,000 for charity!!
- And, yes, there will be a 2021 #DFIRFit challenge
  - Well, if we make it to 2021







## What is this RDP Thing?

- "Remote Desktop Protocol (RDP) is a proprietary protocol developed by Microsoft, which provides a user with a graphical interface to connect to another computer over a network connection"
  - This means someone can do stuff with things on another computer, whether it is in the next room or halfway around the world



## Does That Have To Do With



## My Investigations?

- Lateral movement in an environment
- Remote connection(s) to known/suspected malicious systems
- Unauthorized access
- Ransomware investigations







## How I Got Interested in This Topic

- Working what seemed to be a typical ransomware case
  - YARC
- This particular attacker actually cleaned up after themselves
  - Cleared Event Logs
  - Cleared "Recent" data

• This made answering the usual questions (who, what, how, when, data access, data exfil, etc, EXTREMELY difficult)





## How I Got Interested in This Topic

- Fortunately, the attacker did not clean up the RDP Bitmap Cache files
  - Since didn't have much else to go on, this was at least evidence of "something had happened"







## WTF is RDP Bitmap Cache?

 Let's visit the source ( <u>https://docs.microsoft.com/en-us/openspecs/windows\_protocols/ms-rdpbcgr</u> /2da3e165-d1ba-4b65-8909-7a0f7f858d69 )

"A Persistent Bitmap Cache is a store that contains bitmap images that were sent to the client by using the Cache Bitmap (Revision 2) Secondary Drawing Order ([MS-RDPEGDI] section 2.2.2.2.1.2.3). Unlike the Bitmap Caches described in section 3.2.1.13, Persistent Bitmap Caches are not bound to the lifetime of a given RDP connection and their contents are persisted even after the RDP connection is closed."





### Yeah, That Doesn't Help

Okay, that admittedly was a lot

While it is not technically 100% accurate, a better way to think of it is kind of like taking snapshots of the entire screen during an RDP session, which are written to disk on the endpoint that the RDP session originated from







## Oh, That Is Better, Thank You!

 The location of the RDP Bitmap Cache files has shifted over the years, but for the most part they can be found under the path "%USERPROFILE%\AppData\Local\Microsoft\Terminal Server Client\Cache\"





#### More Technical Details

- On older systems, you will usually have a file with a .bmc extension
- Windows 7 and newer systems, you will likely see files that are named "Cache###.bin" (these are incrementally numbered starting at 0000)
- Both file types contain what are essentially small chunks of screenshots that are saved of the remote desktop



# More Reading (AFTER This Presentation, please!)



- <a href="https://www.allthingsdfir.com/do-you-even-bitmap-cache-bro/">https://www.allthingsdfir.com/do-you-even-bitmap-cache-bro/</a>
- https://countuponsecurity.com/tag/rdp-bitmap-cache/
- https://cbtgeeks.com/2018/05/22/digital-forensics-on-rdp-cache/
- RDP Cache Forensics 13Cubed: <u>https://www.youtube.com/watch?v=NnEOk5-Dstw</u>
- https://www.brimorlabsblog.com/2019/06/phinally-using-photoshop -to-phacilitate.html (Hey, that one is mine!)



# Well Brian, We Can Read. So Why



#### Are You Here?

- Well, part of the reason is because, like everything else that I do, I want to find an easier way to get usable information from this data source
- I very much enjoy OSDFCON every year, and this is an open source project, so it makes sense
  - Although this time, it is virtual. Which means not watching giant robots fight, while enjoying a stack of pizzas approximately one Sarah Edwards high, with ~20 of my closest friends





- Step 1: Extract the data from the RDP Bitmap Cache file(s)
  - I always use the -b flag ... but that is up to you
- In my opinion, best current option for this is the Python script from the ANSSI (agence nationale la sécurité des systèmes d'information) github repository
  - https://github.com/ANSSI-FR/bmc-toolsNote: Use Python v2



## You Put Files In, You Get Usable



### Data Out

- Made a small update to the script to fix a bug
  - The data within the header, referencing file size, is off by 4 bytes (four bytes too long)
    - Most likely counted the "BM" file header (2 bytes) plus hex representation of file size (2 bytes), twice
  - Opened bug request
- Until it is addressed, use this one:
   https://github.com/brimorlabs/rdpieces/blob/master/modifie d-bmc-tools.py





- Alternatively, a PowerShell option is available if you wish to use it.
  - https://github.com/gtworek/PSBits/blob/master/DFIR/DecodeRDPCache.ps1
  - Note: My solution does not currently support the output from this script (if enough interest/requests are made, I can work on building support for this output too though)



# You Put Files In, You Get Usable



### Data Out

Step 2: We now have results. Folder structure probably looks like this:





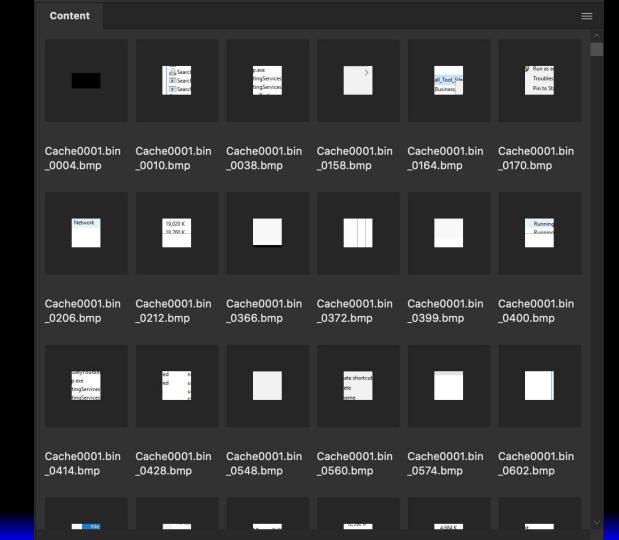
Name ^	•	Date Modified	Size	Kind
Cache0000		Jun 17, 2019 at 3:45 PM		 Folder
Cache0001		Jun 16, 2019 at 11:56 PM		 Folder
Cache0002		Jun 17, 2019 at 12:13 AM		 Folder





Name	^	Date Modified	Size	Kind
<b>a</b>	Cache0000.bin_0000.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image
	Cache0000.bin_0001.bmp	Jun 16, 2019 at 11:01 PM	9 KB	Windows BMP image
	Cache0000.bin_0002.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image
	Cache0000.bin_0003.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image
	Cache0000.bin_0004.bmp	Jun 16, 2019 at 11:01 PM	9 KB	Windows BMP image
	Cache0000.bin_0005.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image
	Cache0000.bin_0006.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image
	Cache0000.bin_0007.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image
	Cache0000.bin_0008.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image
	Cache0000.bin_0009.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image
	Cache0000.bin_0010.bmp	Jun 16, 2019 at 11:01 PM	17 KB	Windows BMP image

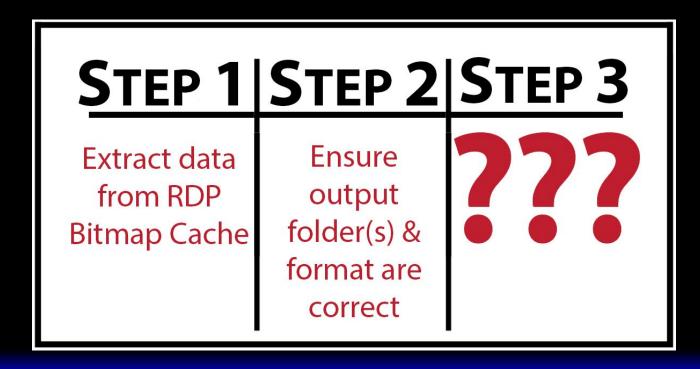








I think this is how this all works?





## You Put Files In, You Get Usable



### Data Out

- Step 4: Now you have a whole bunch of bitmap images (usually 6000+) that are 64 x 64\*, and one large bitmap file with all of the tiles lined up (see next slide)
  - You can now manually rearrange the individual bitmap images, in hopes of "reconstructing" screen shots that are automatically taken, and stored, during the RDP session
  - This is a challenging, and tedious task

\*While a majority are 64 x 64, not all of the images are actually that size. Which makes reconstruction even trickier.





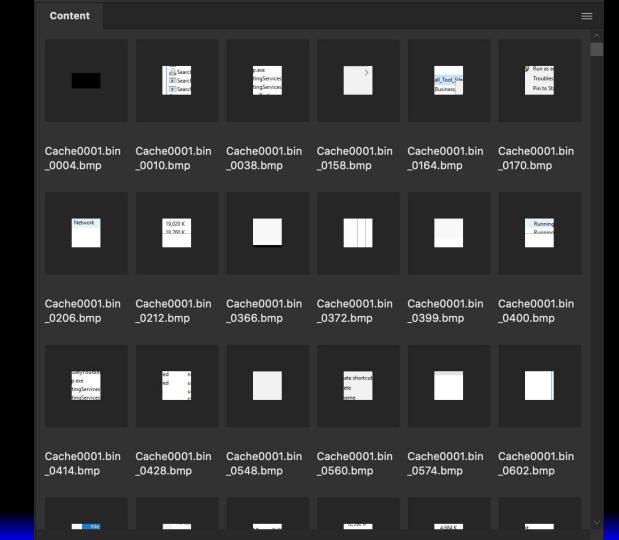








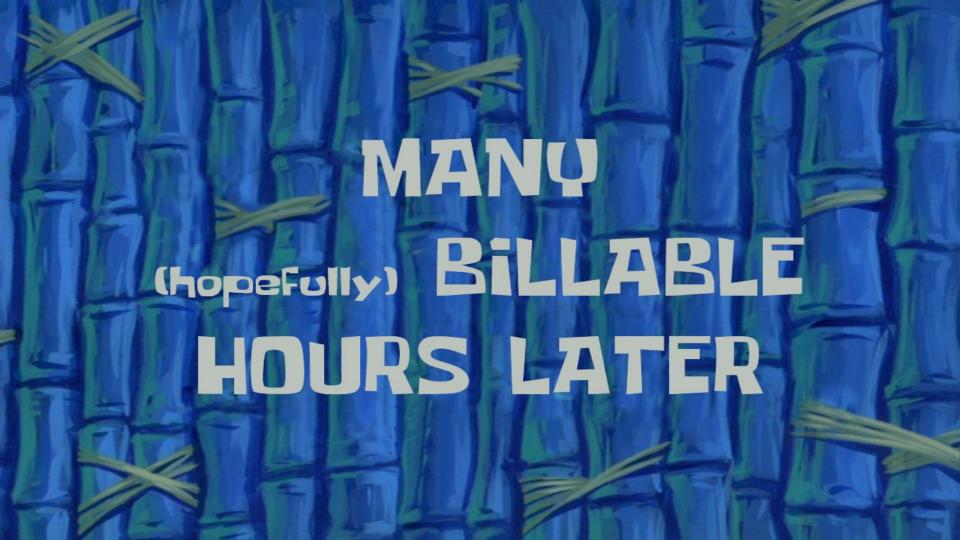




















# You Put Files In, You Get Usable Data Out - Manual Reconstruction





# You Put Files In, You Get Usable



### **Data Out**

- On average, it takes between 20-40 hours to go through and manually rebuild RDP Bitmap Cache data
  - Fine if you have the time (or cough cough billable hours) to do that
- Wanted to make an easier way to at least make slices, and focus on individual slices rather than rebuilding the entire picture



First thought was

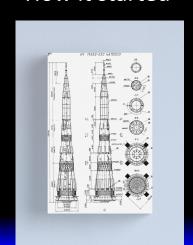






• Started mapping out math, data visualization, statistics, etc. that I thought would be needed

How it started



How it's going





# You Put Files In, You Get Usable



#### Data Out

- Enter imagemagick, which as it turns out, does almost everything that I was hoping to find out, and more, already
  - https://imagemagick.org/index.php





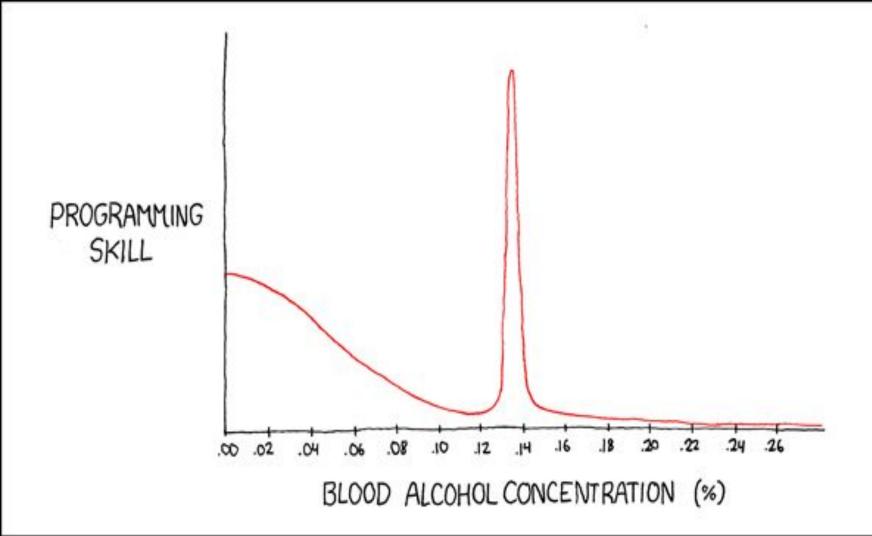


- Now that I found where the mathiness would come from, I
  had to work on ensuring that my formulas worked fairly well,
  were broad enough to capture less than ideal circumstances,
  but at the same time, didn't accidentally match too much
- So relieved that I wouldn't have to do terribly complex data manipulation
  - However, it is worth noting that Python absolutely sucks for doing even moderately advanced mathiness, but Perl handles it all like a champ. Long Live Perl!





- The next hurdle was deciding "how" to do this most efficiently
  - Thankfully, my photography hobby (which I do not focus on nearly enough anymore) came into play
  - Alcohol helped too





# You Put Files In, You Get Usable Data Out



 When matching puzzle pieces, you generally look for shapes that go together ... and the shapes are determined by the edges ...

... hmmm ... this line of thinking might actually take me somewhere. Maybe. Possibly.

Probably need another drink ... I mean, RDP Bitmap Cache inspiration juice





- So, maybe if I just take the edges of each slice, and figure out how many colors, the color variance/standard deviation, and some file name spatial awareness, maybe I could generate some useful data
- After trial and error, deciding that the edge should be 5 pixels in width/height, depending on if we are matching left/right or top/bottom
  - It's not perfect, but it is at least a decent solution!





- Used imagemagick to make a total of four new files (filename + L/R/T/B) for each 5x64 or 64x5 slice
  - Code has been updated to account for numbers less than
     64 now too, since that's how these work (apparently)
- Pushed the resulting mathiness to a SQLite database that is in memory (for returning faster results)
- The formula can (and undoubtedly will) evolve over time, but it's much easier building SQLite queries than computing mathematical statistics of files!





### Introducing: RDPieces.pl

- Perl script that automates everything I just talked about
  - Runs cross platform (Windows, macOS, \*nix)
    - macOS/\*nix may require some additional modules
    - On Windows, use Strawberry Perl (the best Perl)
  - Requires imagemagick to be installed
  - Script cleans up after itself, deleting temp data directory
- At some point, might make a cool logo for it
  - If large scale ransomware cases ever stop
    - Well, at least slows down





- In my testing, there are roughly 400 results to review per bitmap cache (compared to ~6400 files)
- Put limits on the maximum/minimum size of the slices, because that is how math works
- Script also saves a folder with the rebuilt bitmap image, and the original files used to build the bitmap image, if you want to manually manipulate the files a bit.
  - Much easier than doing it all manually



\*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6082.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6083.bmp \*\*\*\*\* NOTICE: Histogram contains only 4 color(s), moving to next file \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6084.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6085.bmp \*\*\*\*\* NOTICE: Histogram contains only 5 color(s), moving to next file \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6086.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6087.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6088.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6089.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6090.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6091.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6092.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6093.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6094.bmp \*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC -IP-Test/Cache0000.bin 6095.bmp \*\*\*\*\* NOTICE: Histogram contains only 2 color(s), moving to next file

```
***** Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC
-IP-Test/Cache0000.bin_6084.bmp

***** Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC
-IP-Test/Cache0000.bin_6085.bmp

***** NOTICE: Histogram contains only 5 color(s), moving to next file

***** Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC
-IP-Test/Cache0000.bin_6086.bmp

***** Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC
```

\*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC

\*\*\*\*\* NOTICE: Histogram contains only 4 color(s), moving to next file

-IP-Test/Cache0000.bin 6083.bmp

-IP-Test/Cache0000.bin 6087.bmp

\*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC
-IP-Test/Cache0000.bin\_6088.bmp

\*\*\*\*\* Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC
-IP-Test/Cache0000.bin 6089.bmp





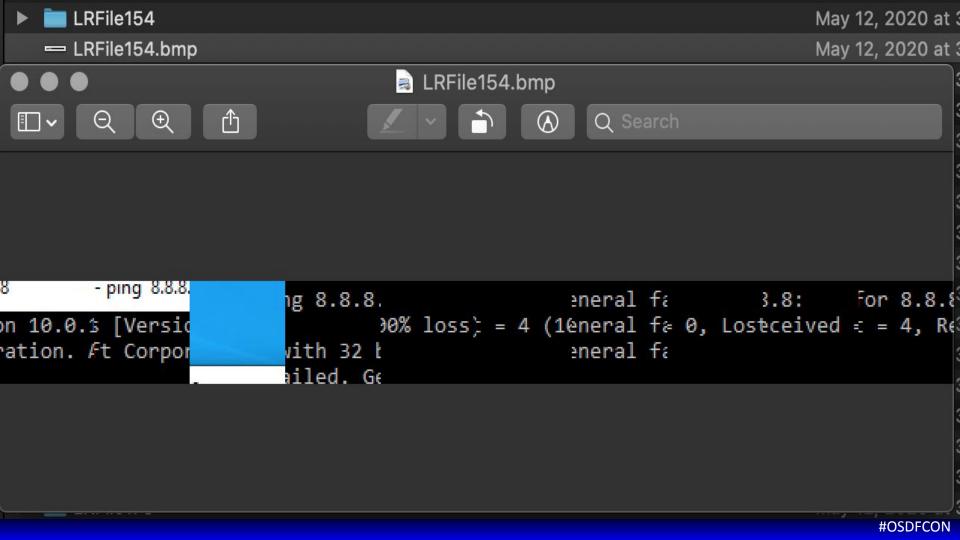
## Example of RDPieces.pl running

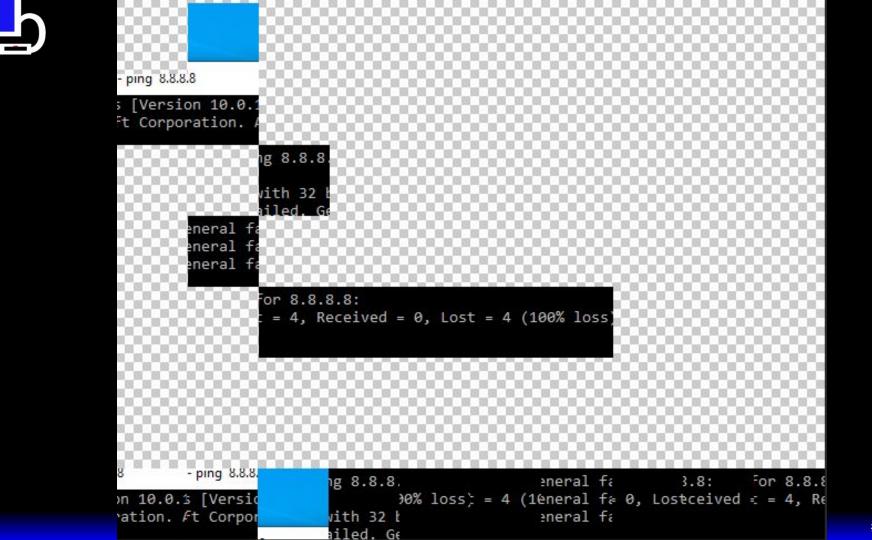
```
***** Processing /Users/brimorlabs/Desktop/RDP-Bitmap-Cache-Research/Testing/UAC-IP-Test/Cache0000.bin_6100.bmp
```

Now deleting the directory /Users/brimorlabs/Desktop/foobar/Data/

A total of 12 files have been copied

The script took 00:00:24 to complete









#### Data Out

 Because we have the original files that the slice was comprised of, we can then go back and try to rebuild a more complete picture with other slices and/or images









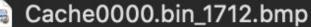




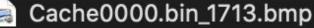








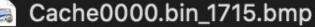












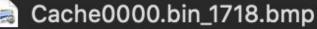




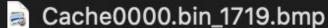




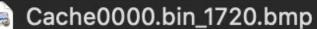








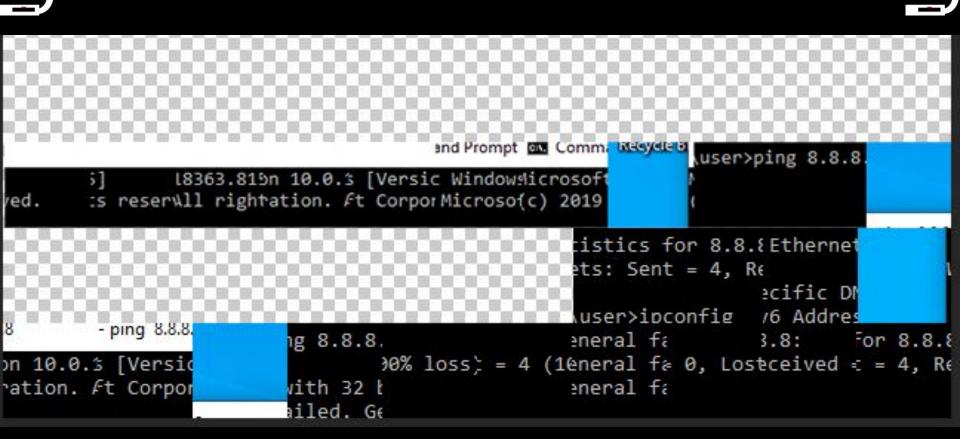


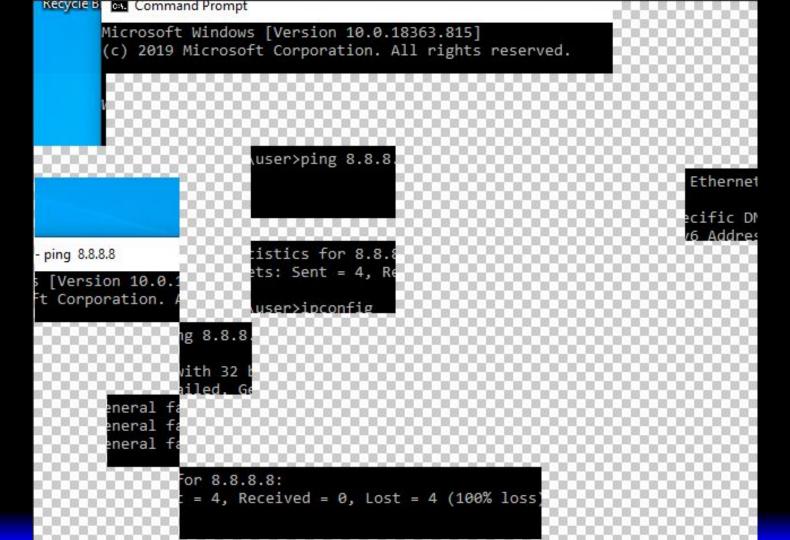


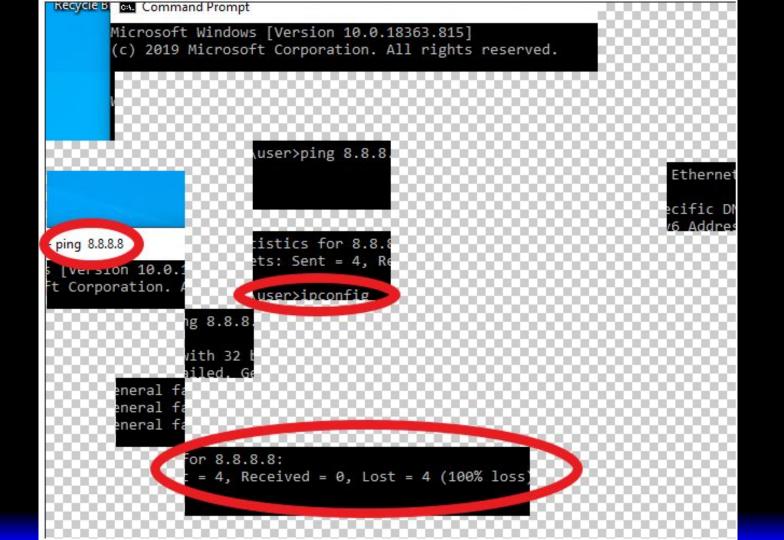














# You Put Files In, You Get Usable Data Out

**\_** 

- Hey, that isn't too shabby, right?
- We can see that
  - Windows command prompt was used
  - User ran commands "ping" and "ipconfig"





# You Put Files In, You Get Usable Data Out - The Next Generation



- This is going to be a continuing project
  - Very much welcome feedback, comments, thoughts on ways to improve it
- My only caveat is that I want to keep this project entirely open source.
  - If Microsoft will not release technical details of how they are doing/rebuilding it, at least we as a community can band together to try to come up with a solution!



## Cool Story Bro, Where Can I Get It?

- You can download the Perl script here:
  - https://github.com/brimorlabs/rdpieces
- Again, my only caveat is that I want to keep this project entirely open source
  - I am sure there are different,
     and probably better,
     ways to perform mathiness
  - Sharing is caring



#### **BriMor Labs**

#### RDPieces.pl

This script will parse extracted RDP Bitmap Cache directory(ies) and attempt to rebuild some of the screenshots automatically. A user is required to extract the bmp files already, best done by using the script from https://github.com/ANSSI-FR/bmc-tools

Usage example: rdpieces.pl -source "RDPBitmapFiles" -output "Rebuilt Images"

#### SUPPORTED PLATFORMS:

- Windows
- macOS
- \*nix

#### REQUIREMENTS:

- Needs output from ANSSI bmc-tools Python script (use Python 2)
- · May require some additional Perl modules
- · On Windows, highly suggest using Strawberry Perl
- Users must have Imagemagick installed on their system, as that program does most of the heavy lifting. Please visit https://imagemagick.org and install imagemagick if you have not done so already





### Data Out

Heather (@LitMoose) summed it up best:



"It is like putting together an adult jigsaw puzzle, but for forensic analysts"

She also said something to the effect of "it's kind of relaxing", which makes me question things about her





### Naughty By Nature said it best ...







#### Questions?



#### **Brian Moran**

Twitter: @brianjmoran

Email: brian@brimorlabs.com

Wear a mask
Wash your hands
Practice social distancing
Avoid large (in-person) gatherings