Quick important information

- Passwords are most commonly cracked using GPUs
- Data for password cracking is hard to come by, but I think we are able just to use cryptocurrency mining data instead, which is in abundance.
- FLOPS (Floating Point Operations per second) is the number of calculations a device can make in a second, also known as math problems.
- HPS (Hashes per second) is the number of guesses (password guesses) a device can make on a specific algorithm per second
- Passwords hashed in specific algorithms must be decrypted using the same algorithm it was hashed in

We may be able to create a database of popular graphics cards and their FLOPS then find out how many FLOPS it takes for every hash in every algorithm.

Most common password hashing algorithms:

- MD5 (Second most common, easy to crack)
- SHA-1, SHA-256, SHA384 (Most Common in General, somewhat easy to crack)
- BCrypt (Best for security, slowest to crack)
- Argon (Good security, slow to crack)

If we are able to use GPU mining data, it will be substantially easier to implement a lot of the advanced features.

Calculating FLOPS of GPUs

FP64:

(SMs * GHz) * 4 = FP64 Performance in GFLOPS
SMs = Stream Multiprocessors
GHz = Frequency of individual CUDA Cores
FP64 = (Double) Precision Floating-Point Format

An example would be,
3090Ti has 84 Stream Multiprocessors
3090Ti has a boost clock of 1.86GHz
(84 * 1.86) * 4 = 624.96 (FP64) GFLOPS
Real-world result: 625.0 GFLOPS

FP32:

> Double Precision Flops Rating = Clock frequency x CUDA Cores x 2 x Clock cycles
> X = Clock Cycles
> I'm unsure of what the relevance of Clock Cycles are in this case but it doesn't seem to be relevant as I get the correct answer without it.
> **((GHz * Core) * 2) * X**
>
> An example would be,
> 1.86 (Boost clock speed) * 10752 (Core count) = 19,998.72
> 19,998.72 * 2 = 39,997.44 TFLOPS
>
> **Real-world result:** 40.00 TFLOPS

I believe we should use FP64 as it's more precise in the number of decimals it has… Though this is irrelevant if we are just randomly guessing. Eh, I found this article saying most hashing algorithms do not use floating-point operations and only use integers, so the data above may be irrelevant. Maybe we have to use IPS (Instructions per second instead)?

Real-world HASHCAT 3090-Ti benchmarks algorithm's results

MD5 - 65079.1 MH/s
MD5 (pass + salt) - 66252.7 MH/s
MD5 (salt + pass) - 37131.6 MH/s
SHA-1 - 22757.6 MH/s
SHA-256 (pass + salt) - 9746.6 MH/s

And a lot more I did not include because there was so much.
Data collected from Hashcat v6.1.1 benchmark on the Nvidia RTX 3090 · GitHub

Sources:

https://www.hivesystems.io/blog/are-your-passwords-in-the-green
https://developer.okta.com/blog/2019/07/29/hashing-techniques-for-password-storage
https://www.tomshardware.com/news/eight-rtx-4090s-can-break-passwords-in-under-an-hour
https://www.azcalculator.com/calc/GPU-gflops-calculator.php
https://gist.github.com/Chick3nman/e4fcee00cb6d82874dace72106d73fef