

Books

- [Hash Crack: Password Cracking Manual \(v3\)](#) - Password Cracking Manual v3 is an expanded reference guide for password recovery (cracking) methods, tools, and analysis techniques.

Cloud

- [Cloud_crack](#) - Crack passwords using Terraform and AWS.
- [Cloudcat](#) - A script to automate the creation of cloud infrastructure for hash cracking.
- [Cloudstomp](#) - Automated deployment of instances on EC2 via plugin for high CPU/GPU applications at the lowest price.
- [Cloudtopolis](#) - A tool that facilitates the installation and provisioning of Hashtopolis on the Google Cloud Shell platform, quickly and completely unattended (and also, free!).
- [NPK](#) - NPK is a distributed hash-cracking platform built entirely of serverless components in AWS including Cognito, DynamoDB, and S3.
- [Penglabs](#) - Abuse of Google Colab for cracking hashes.
- [Rook](#) - Automates the creation of AWS p3 instances for use in GPU-based password cracking.

Conversion

- [7z2hashcat](#) - Extract information from password-protected .7z archives (and .sfx files) such that you can crack these "hashes" with hashcat.
- [MacinHash](#) - Convert macOS plist password file to hash file for password crackers.
- [NetNTLM-Hashcat](#) - Converts John The Ripper/Cain format hashes (singular, or in bulk) to HashCat compatible hash format.
- [Rubeus-to-Hashcat](#) - Converts / formats Rubeus kerberoasting output into hashcat readable format.
- [WINHELLO2hashcat](#) - With this tool one can extract the "hash" from a WINDOWS HELLO PIN. This hash can be cracked with Hashcat.
- [bitwarden2hashcat](#) - A tool that converts Bitwarden's data into a hashcat-suitable hash.
- [hc_to_7z](#) - Convert 7-Zip hashcat hashes back to 7z archives.
- [hcxtools](#) - Portable solution for conversion of cap/pcap/pcapng (gz compressed) WiFi dump files to hashcat formats.
- [itunes_backup2hashcat](#) - Extract the information needed from the Manifest.plist files to convert it to hashes compatible with hashcat.
- [mongodb2hashcat](#) - Extract hashes from the MongoDB database server to a hash format that hashcat accepts: -m 24100 (SCRAM-SHA-1) or -m 24200 (SCRAM-SHA-256).

Hashcat

[Hashcat](#) is the "World's fastest and most advanced password recovery utility." The following are projects directly related to Hashcat in one way or another.

- [Autocrack](#) - A set of client and server tools for automatically, and lightly automatically cracking hashes.
- [docker-hashcat](#) - Latest hashcat docker for Ubuntu 18.04 CUDA, OpenCL, and POCL.
- [hashcat.launcher](#) - Hashcat.launcher is a cross-platform GUI app that run and control hashcat.
- [Hashcat-Stuffs](#) - Collection of hashcat lists and things.
- [hashcat-utils](#) - Small utilities that are useful in advanced password cracking.
- [Hashfilter](#) - Read a hashcat potfile and parse different types into a sqlite database.
- [known_hosts-hashcat](#) - A guide and tool for cracking ssh known_hosts files with hashcat.
- [pyhashcat](#) - Python C API binding to libhashcat.

Automation

- [autocrack](#) - Hashcat wrapper to help automate the cracking process.
- [hat](#) - An Automated Hashcat Tool for common wordlists and rules to speed up the process of cracking hashes during engagements.
- [hate_crack](#) - A tool for automating cracking methodologies through Hashcat from the TrustedSec team.
- [Naive hashcat](#) - Naive hashcat is a plug-and-play script that is pre-configured with naive, empirically-tested, "good enough" parameters/attack types.

Distributed cracking

- [CrackLord](#) - Queue and resource system for cracking passwords.
- [fitcrack](#) - A hashcat-based distributed password cracking system.
- [Hashstation](#) - Hashstation is a BOINC-based distributed password cracking system with a built-in web interface.
- [Hashtopolis](#) - A multi-platform client-server tool for distributing hashcat tasks to multiple computers.
- [Kraken](#) - A multi-platform distributed brute-force password cracking system.

Rules

- [clem9669 rules](#) - Rule for hashcat or john.
- [hashcat rules collection](#) - Probably the largest collection of hashcat rules out there.
- [Hob0Rules](#) - Password cracking rules for Hashcat based on statistics and industry patterns.
- [Kaonashi](#) - Wordlist, rules and masks from Kaonashi project (RootedCON 2019).
- [nsa-rules](#) - Password cracking rules and masks for hashcat generated from cracked passwords.

- [nyxgeek-rules](#) - Custom password cracking rules for Hashcat and John the Ripper.
- [OneRuleToRuleThemAll](#) - "One rule to crack all passwords. or atleast we hope so."
- [OneRuleToRuleThemStill](#) - "A revamped and updated version of my original OneRuleToRuleThemAll hashcat rule."
- [pantagrule](#) - Large hashcat rulesets generated from real-world compromised passwords.

Rule tools

- [duprule](#) - Detect & filter duplicate hashcat rules.
- [ruleprocessorY](#) - A next-gen Rule processor with complex multibyte character support built to support Hashcat.

Web interfaces

- [crackerjack](#) - CrackerJack is a Web GUI for Hashcat developed in Python.
- [CrackQ](#) - A Python Hashcat cracking queue system.
- [hashpass](#) - Hash cracking WebApp & Server for hashcat.
- [Hashview](#) - A web front-end for password cracking and analytics.
- [Wavecrack](#) - Wavestone's web interface for password cracking with hashcat.
- [WebHashCat](#) - WebHashcat is a very simple but efficient web interface for hashcat password cracking tool.

John the Ripper

[John the Ripper](#) is "an Open Source password security auditing and password recovery tool available for many operating systems." The following are projects directly related to John the Ripper in one way or another.

- [BitCracker](#) - BitCracker is the first open source password cracking tool for memory units encrypted with BitLocker.
- [johnny](#) - GUI frontend to John the Ripper.

Misc

- [hashgen](#) - Hashgen is a simple yet very fast CLI hash generator written in Go and cross compiled for Linux, Windows & Mac.
- [Name That Hash](#) - Don't know what type of hash it is? Name That Hash will name that hash type! Identify MD5, SHA256 and 300+ other hashes. Comes with a neat web app.

Websites

Communities

- [hashcat Forum](#) - Forum by the developers of hashcat.

- [Hashmob](#) - A growing password recovery community aimed towards being a center point of collaboration for cryptography enthusiasts. Huge wordlist collection and a lookup service too.
- [Hashkiller Forum](#) - A password cracking forum with over 20,000 registered users.

Lookup services

- [CMD5](#) - Provides online MD5 / sha1/ mysql / sha256 encryption and decryption services.
- [CrackStation](#) - Free hash lookup service supplying wordlists as well.
- [gohashmob](#) - Go CLI app to quickly lookup hashes using the HashMob API.
- [Hashes.com](#) - A hash lookup service with paid features.
- [Hashkiller](#) - A hash lookup service with a forum.
- [Online Hash Crack](#) - Cloud password recovery service.

Wordlist tools

Tools for analyzing, generating and manipulating wordlists.

Analysis

- [PACK](#) - A collection of utilities developed to aid in analysis of password lists in order to enhance password cracking through pattern detection of masks, rules, character-sets and other password characteristics.
- [password-smelter](#) - Ingests passwords from hashcat, etc. and outputs to HTML, Markdown, XLSX, PNG, JSON. Dark and light themes supported. Compliments password-stretcher.
- [password-stretcher](#) - Generate "disgusting quantities" of passwords from websites, files, or stdin. Compliments password-smelter.
- [pcfg_cracker](#) - This project uses machine learning to identify password creation habits of users.
- [Pipal](#) - THE password analyser.
- [Graphcat](#) - Generate graphs and charts based on password cracking result.

Generation/Manipulation

- [accent_permutator](#) - A tool to transform characters from ASCII / UTF-8 to accented characters such as "o" to "ò".
- [anew](#) - Append lines from stdin to a file, but only if they don't already appear in the file. Outputs new lines to stdout too, making it a bit like a tee -a that removes duplicates.
- [bopscrk](#) - Generate smart and powerful wordlists for targeted attacks. Includes song lyrics fetching and different transforms.
- [common-substr](#) - Simple tool to extract the most common substrings from an input text. Built for password cracking.

- [Crunch](#) - Crunch is a wordlist generator where you can specify a standard character set or a character set you specify. Crunch can generate all possible combinations and permutations.
- [CUPP](#) - A tool that lets you generate wordlists by user profiling data such as birthday, nickname, address, name of a pet or relative etc.
- [duplicut](#) - Remove duplicates from MASSIVE wordlist, without sorting it (for dictionary-based password cracking).
- [Gorilla](#) - Tool for generating wordlists or extending an existing one using mutations.
- [Gramify](#) - Create n-grams of wordlists based on words, characters, or charsets to use in offline password attacks and data analysis.
- [Elpscrk](#) - Elpscrk is like cupp, but it's based on permutations and statistics while being memory efficient.
- [Keyboard-Walk-Generators](#) - Generate Keyboard Walk Dictionaries for cracking.
- [kwprocessor](#) - Advanced keyboard-walk generator with configurable basechars, keymap and routes.
- [maskcat](#) - Utility tool for Hashcat Masks and Password Cracking.
- [maskprocessor](#) - High-performance word generator with a per-position configurable charset.
- [maskuni](#) - A standalone fast word generator in the spirit of hashcat's mask generator with unicode support.
- [Mentalist](#) - Mentalist is a graphical tool for custom wordlist generation. It utilizes common human paradigms for constructing passwords and can output the full wordlist as well as rules compatible with Hashcat and John the Ripper.
- [PTT](#) - The Password Transformation Tool (ptt) is a versatile utility designed for password cracking. It facilitates the creation of custom rules and transformations, as well as the generation of wordlists. This tool supports processing data from files, URLs, and standard input, streamlining cracking workflows.
- [Phraser](#) - Phraser is a phrase generator using n-grams and Markov chains to generate phrases for passphrase cracking.
- [princeprocessor](#) - Standalone password candidate generator using the PRINCE algorithm.
- [Rephraser](#) - A Python-based reimagining of Phraser using Markov-chains for linguistically-correct password cracking.
- [Rling](#) - RLI Next Gen (Rling), a faster multi-threaded, feature rich alternative to rli found in hashcat utilities.
- [statsprocessor](#) - Word generator based on per-position markov-chains.
- [StringZilla](#) - Fastest string sort, search, split, and shuffle for long strings and multi-gigabyte files in Python and C.
- [TTPassGen](#) - Flexible and scriptable password dictionary generator which supports brute-force, combination, complex rule modes etc.
- [token-reverser](#) - Words list generator to crack security tokens.
- [WikiRaider](#) - WikiRaider enables you to generate wordlists based on country specific databases of Wikipedia.

Wordlists

Language specific

- [Albanian wordlist](#) - A mix of names, last names and some albanian literature.
- [Danish Phone Wordlist Generator](#) - This tool can generate wordlists of Danish phone numbers by area and/or usage (Mobile, landline etc.) Useful for password cracking or fuzzing Danish targets.
- [Danish Wordlists](#) - Collection of danish wordlists for cracking danish passwords.
- [French Wordlists](#) - This project aim to provide french word list about everything a person could use as a base password.

Other

- [Packet Storm Wordlists](#) - A substantial collection of different wordlists in multiple languages.
- [Rocktastic](#) - Includes many permutations of passwords and patterns that have been observed in the wild.
- [RockYou2021](#) - RockYou2021.txt is a MASSIVE WORDLIST compiled of various other wordlists.
- [WeakPass](#) - Collection of large wordlists.

Specific file formats

PDF

- [pdfrip](#) - A multi-threaded PDF password cracking utility equipped with commonly encountered password format builders and dictionary attacks.

JKS

- [JKS private key cracker](#) - Cracking passwords of private key entries in a JKS file.

ZIP

- [bkcrack](#) - Crack legacy zip encryption with Biham and Kocher's known plaintext attack.
- [frackzip](#) - Small tool for cracking encrypted ZIP archives.

Artificial Intelligence

- [adams](#) - Reducing Bias in Modeling Real-world Password Strength via Deep Learning and Dynamic Dictionaries.
- [neural network cracking](#) - Code for cracking passwords with neural networks.
- [RNN-Passwords](#) - Using the char-rnn to learn and guess passwords.

- [rulesfinder](#) - This tool finds efficient password mangling rules (for John the Ripper or Hashcat) for a given dictionary and a list of passwords.
- [PassGPT](#) - PassGPT is a GPT-2 model trained from scratch on password leaks.

Research

Articles and Blog Posts

- [Optimizing Wordlists with Masks](#)
- [Purple Rain Attack - Password Cracking With Random Generation](#)
- [Smashing Hashes with Token Swapping Attacks](#)

Papers

- [Generating Optimized Guessing Candidates toward Better Password Cracking from Multi-Dictionaries Using Relativistic GAN \(2020\)](#)
- [GENPass: A General Deep Learning Model for Password Guessing with PCFG Rules and Adversarial Generation \(2018\)](#)
- [Password Cracking Using Probabilistic Context-Free Grammars \(2009\)](#)
- [Reducing Bias in Modeling Real-world Password Strength via Deep Learning and Dynamic Dictionaries \(2020\)](#)
- [Fast, Lean, and Accurate: Modeling Password Guessability Using Neural Networks \(2016\)](#)
- [PassGAN: A Deep Learning Approach for Password Guessing \(2017\)](#)
- [PassGPT: Password Modeling and \(Guided\) Generation with LLMs](#)

Talks

- [BsidesKY2023 - Leveling Up Password Attacks with Breach Data](#)
- [DEF CON Safe Mode Password Village - Getting Started with Hashcat](#)
- [DEF CON Safe Mode Password Village - Jeremi Gosney - Cracking at Extreme Scale](#)
- [DEF CON 28 Safe Mode Password Village – 'Let's Crack RockYou Without Using rockyou.txt'](#)
- [SecTor 2019 - Will Hunt - Hashes, Hashes Everywhere, But All I See Is Plaintext](#)
- [Tailored, Machine Learning-driven Password Guessing Attacks and Mitigation at DefCamp](#)
- [UNHash - Methods for better password cracking](#)
- [USENIX Security '21 - Reducing Bias in Modeling Real-world Password Strength via Deep Learning and Dynamic Dictionaries](#)
- [USENIX Security '16 - Fast, Lean, and Accurate: Modeling Password Guessability Using Neural Networks](#)