

ALGORYTMY W ŁAMANIU HASIEL

Kunegunda Kosek 59280

WPROWADZENIE

- **Definicja łamania haseł:** Proces odzyskiwania haseł z przechowywanych danych lub ich odgadywania w celu uzyskania nieautoryzowanego dostępu.
- **Znaczenie bezpieczeństwa haseł:** Chroni przed nieautoryzowanym dostępem do systemów i danych.
- **Cel prezentacji:** Przedstawienie metod łamania haseł oraz sposobów ochrony przed nimi.



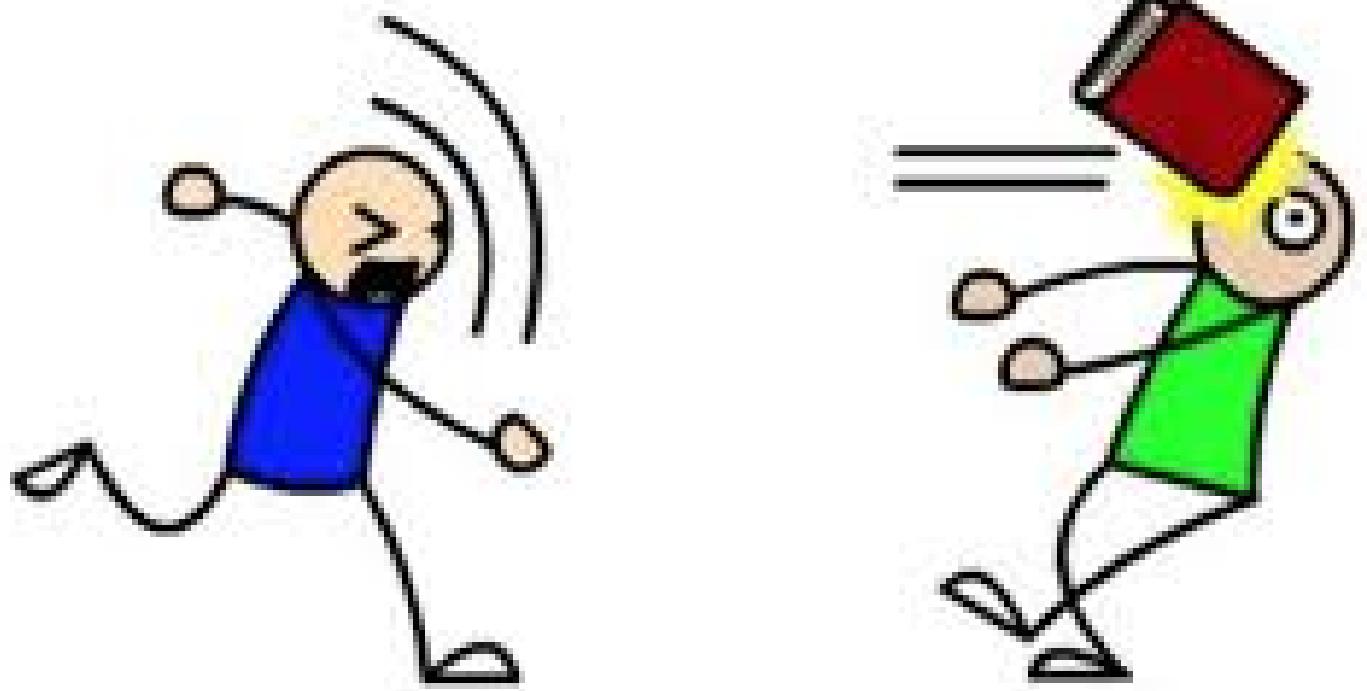
PODSTAWOWE POJĘCIA

- **Hasło:** Sekwencja znaków używana do uwierzytelniania użytkownika.
- **Hashowanie:** Proces przekształcania hasła w unikalny ciąg znaków o stałej długości.
- **Sól (salt):** Losowy ciąg dodawany do hasła przed hashowaniem w celu zwiększenia bezpieczeństwa.
- **Funkcje skrótu:** Algorytmy przekształcające dane wejściowe w skrót o stałej długości.

METODY ŁAMANIA HASEŁ

- **Atak słownikowy:** Wykorzystanie listy popularnych haseł do odgadnięcia właściwego.
- **Atak brute-force:** Próba wszystkich możliwych kombinacji znaków.
- **Atak hybrydowy:** Połączenie ataku słownikowego z modyfikacjami haseł.
- **Atak z użyciem tęczowych tablic:** Wykorzystanie prekomputowanych tablic skrótów do szybkiego odwracania funkcji hashujących.

DICTIONARY ATTACK!



ATAK SŁOWNIKOWY

- **Opis metody:** Sprawdzanie haseł z wcześniejszej przygotowanej listy słów.
- **Wykorzystanie list słów:** Użycie słowników zawierających popularne hasła i ich warianty.
- **Przykłady:** "password", "123456", "qwerty".



0100100
10000101
10110
000110
01001
00101
10110
00110
01001
00101
10110
0110
010

ATAK BRUTE- FORCE

- **Opis metody:** Generowanie i testowanie wszystkich możliwych kombinacji znaków.
- **Czasochłonność:** Zależna od długości i złożoności hasła; dla długich haseł czas łamania może być bardzo długi.
- **Przykład:** Dla 8-znakowego hasła składającego się z małych liter istnieje 26^8 możliwych kombinacji.



ATAK HYBRUDOWY

- **Połączenie metod:** Rozpoczyna się od ataku słownikowego, a następnie stosuje modyfikacje, takie jak dodawanie cyfr czy znaków specjalnych.
- **Modyfikacje słów:** Dodawanie prefiksów, sufiksów, zamiana liter na cyfry (np. 'a' na '4').
- **Przykłady:** "password1", "admin123", "qwerty!" .



TĘCZOWE TABLICE

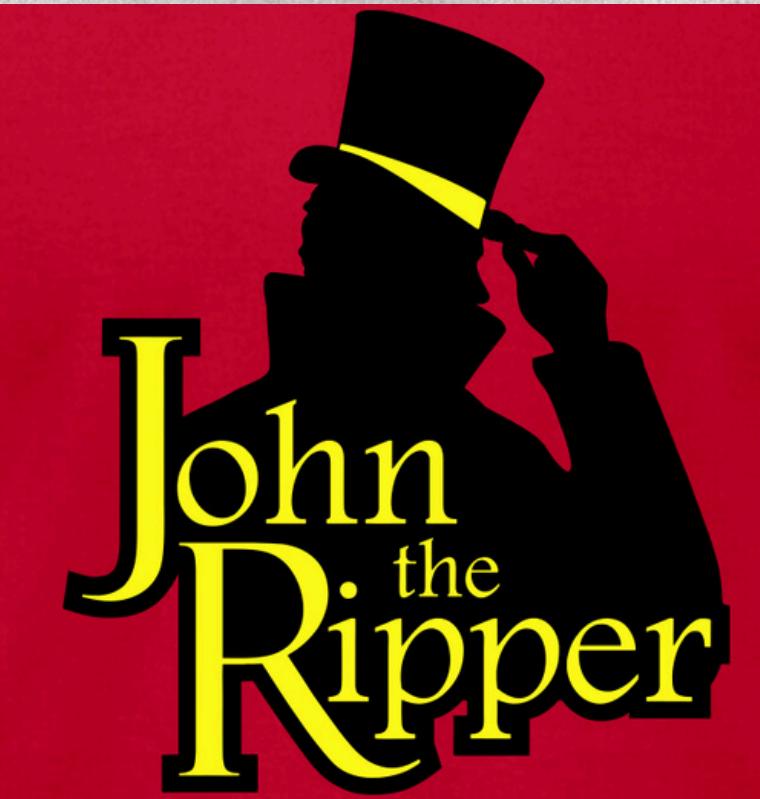
- **Opis metody:** Prekomputowane tablice skrótów dla różnych haseł, umożliwiające szybkie odwracanie funkcji hashujących.
- **Zalety:** Szybkość w porównaniu do tradycyjnych ataków brute-force.
- **Wady:** Wymagają dużej ilości miejsca na przechowywanie tablic; nieskuteczne wobec haseł z unikalną solą.

NARZĘDZIA DO ŁAMANIA HASŁ

- **Hashcat:** Zaawansowane narzędzie wykorzystujące GPU do łamania haseł.
- **John the Ripper:** Popularne narzędzie open-source do łamania haseł.
- **Ophcrack:** Narzędzie wykorzystujące tęczowe tablice do łamania haseł systemu Windows.
- **THC Hydra:** Narzędzie do ataków na protokoły sieciowe z użyciem słowników.



- **Opis:** Wydajne narzędzie do łamania haseł z obsługą wielu algorytmów hashujących.
- **Obsługiwane algorytmy:** MD5, SHA-1, SHA-256, bcrypt i inne.
- **Przykłady użycia:** Łamanie haseł z plików hash z użyciem GPU.



- **Opis:** Wszechstronne narzędzie do łamania haseł, dostępne na wielu platformach.
- **Funkcje:** Obsługa wielu formatów haseł, możliwość dostosowywania ataków.
- **Przykłady użycia:** Łamanie haseł systemów Unix, Windows, baz danych.

HYDRA BRUTEFORCE



- **Opis:** THC Hydra to zaawansowane narzędzie do łamania haseł metodą brute-force, obsługujące wiele protokołów sieciowych.
- **Obsługiwane protokoły:** FTP, SSH, HTTP(S), SMB, POP3, IMAP, MySQL, VNC i wiele innych.

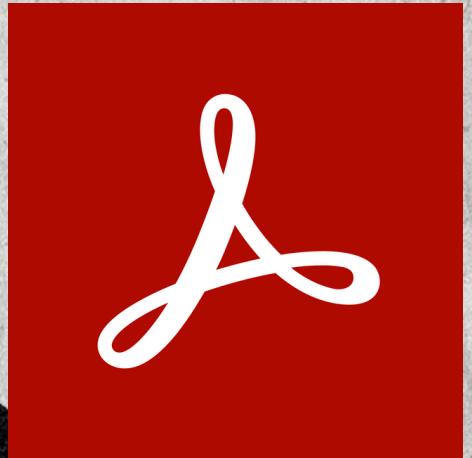
- - Równoległe ataki na wiele celów jednocześnie.
 - Możliwość wykorzystania różnych metod ataku, w tym ataków słownikowych i brute-force.
 - Wsparcie dla uwierzytelniania opartego na formularzach internetowych.
- **Zastosowanie:** Używane przez testerów penetracyjnych do identyfikacji słabych haseł i oceny bezpieczeństwa systemów.
- **Ostrzeżenie:** Należy używać wyłącznie w celach legalnych i za zgodą właścicieli systemów.

PRZYSZŁOŚĆ ŁAMANIA HASEŁ

- **Postęp technologiczny:** Rozwój komputerów kwantowych może znaczco skrócić czas potrzebny na złamanie tradycyjnych algorytmów kryptograficznych.
- **Nowe metody ataków:** Pojawienie się zaawansowanych technik, takich jak ataki oparte na uczeniu maszynowym, może zwiększyć skuteczność łamania haseł.
- **Ewolucja zabezpieczeń:** W odpowiedzi na nowe zagrożenia rozwijane są bardziej zaawansowane metody ochrony, takie jak uwierzytelnianie wieloskładnikowe i biometryczne.

PRZYKŁADY RZECZYWISTYCH

ATAKÓW



- **LinkedIn (2012)**: Wykradzono hasła milionów użytkowników, co uwidoczyło znaczenie stosowania silnych haseł i ich odpowiedniego przechowywania.
- **Yahoo (2013-2014)**: Atakujący uzyskali dostęp do danych ponad miliarda kont, co podkreśliło potrzebę regularnej zmiany haseł i monitorowania naruszeń bezpieczeństwa.
- **Adobe (2013)**: Wycieki danych uwidoczyły znaczenie szyfrowania i bezpiecznego przechowywania informacji uwierzytelniających.

ETYKA I LEGALNOŚĆ ŁAMANIA HASEŁ

- **Zastosowania legalne:** Testy penetracyjne i audyty bezpieczeństwa przeprowadzane za zgodą w celu identyfikacji i naprawy luk w systemach.
- **Konsekwencje nielegalnych działań:** Nieautoryzowane łamanie haseł jest przestępstwem i może prowadzić do surowych sankcji prawnych.
- **Odpowiedzialność specjalistów IT:** Profesjonalisi powinni działać zgodnie z zasadami etyki i obowiązującym prawem, dbając o bezpieczeństwo danych.

REKOMENDACJE DLA UŻYTKOWNIKÓW

- **Tworzenie silnych haseł:** Używanie długich, unikalnych haseł zawierających kombinację liter, cyfr i znaków specjalnych.
- **Regularna zmiana haseł:** Aktualizowanie haseł co określony czas, zwłaszcza po podejrzeniu naruszenia bezpieczeństwa.
- **Unikanie powtórnego użycia haseł:** Nie stosowanie tego samego hasła w różnych serwisach i aplikacjach.

REKOMENDACJE DLA ORGANIZACJI

- **Szkolenia z zakresu bezpieczeństwa:** Edukowanie pracowników na temat znaczenia silnych haseł i rozpoznawania prób phishingu.
- **Wdrażanie polityk haseł:** Ustalanie wymagań dotyczących złożoności i częstotliwości zmiany haseł.
- **Monitorowanie i audyt:** Regularne sprawdzanie systemów pod kątem nieautoryzowanych prób dostępu i potencjalnych luk w zabezpieczeniach.

ALTERNatywne metody uwierzytelniania

- **Uwierzytelnianie dwuskładnikowe (2FA):** Dodanie dodatkowej warstwy zabezpieczeń poprzez wymaganie drugiego elementu uwierzytelniającego, takiego jak kod SMS czy aplikacja uwierzytelniająca.
- **Uwierzytelnianie biometryczne:** Wykorzystanie cech fizycznych użytkownika, takich jak odcisk palca czy rozpoznanie twarzy, do weryfikacji tożsamości.
- **Klucze bezpieczeństwa:** Użycie fizycznych urządzeń, takich jak klucze USB, do uwierzytelniania użytkownika.

WYZWANIA W ZABEZPIECZANIU HASEŁ

- **Czynnik ludzki:** Użytkownicy często wybierają proste hasła lub używają tych samych haseł w wielu miejscach, co zwiększa ryzyko naruszenia bezpieczeństwa.
- **Ewolucja technologii:** Nowe technologie mogą zarówno poprawiać, jak i osłabiać bezpieczeństwo haseł, w zależności od ich zastosowania.
- **Zarządzanie hasłami:** Trudności w bezpiecznym przechowywaniu i zarządzaniu wieloma hasłami w organizacjach.

PODSUMOWANIE

- **Znaczenie silnych haseł:** Silne i unikalne hasła są kluczowe dla ochrony danych osobowych i firmowych.
- **Edukacja i świadomość:** Stałe podnoszenie świadomości na temat zagrożeń i najlepszych praktyk w zakresie bezpieczeństwa haseł jest niezbędne.
- **Ciągłe doskonalenie:** Organizacje i użytkownicy powinni regularnie aktualizować swoje metody zabezpieczeń w odpowiedzi na nowe zagrożenia.

.\hashcat.exe -m 0 -a 0 hashes.txt rockyou.txt

Składnia komendy:

.\hashcat.exe: Uruchamia program Hashcat w bieżącym katalogu.

-m 0: Określa typ hasha jako MD5.

-a 0: Ustawia tryb ataku na prosty atak słownikowy.

hashes.txt: Plik zawierający hashe do złamania.

rockyou.txt: Plik słownika z potencjalnymi hasłami.

Szczegółowe wyjaśnienie:

-m 0: Opcja -m definiuje typ hasha, który chcemy złamać. W tym przypadku 0 odpowiada hashom MD5. Hashcat obsługuje wiele typów hashów, a pełną listę można znaleźć w dokumentacji narzędzia.

-a 0: Opcja -a określa tryb ataku. Wartość 0 oznacza prosty atak słownikowy, w którym Hashcat przegląda każde słowo z podanego słownika i porównuje jego hash z hashami w pliku hashes.txt.

hashes.txt: Ten plik powinien zawierać hashe, które chcemy złamać. Każdy hash powinien być w osobnej linii.

rockyou.txt: Jest to popularny plik słownika zawierający miliony powszechnie używanych haseł. Hashcat używa tego pliku jako źródła potencjalnych haseł do porównania z hashami w hashes.txt.

Przykład działania:

Jeśli plik hashes.txt zawiera hash:

5f4dcc3b5aa765d61d8327deb882cf99

A plik rockyou.txt zawiera hasło "password", Hashcat wygeneruje hash dla "password" i porówna go z hashem w hashes.txt. Ponieważ hash "password" to właśnie 5f4dcc3b5aa765d61d8327deb882cf99, Hashcat zidentyfikuje to hasło jako pasujące.

Dodatkowe informacje:

Typy hashów (-m): Hashcat obsługuje różne typy hashów, takie jak SHA-1, SHA-256, bcrypt i wiele innych. Warto sprawdzić dokumentację Hashcat, aby znaleźć odpowiedni kod dla hasha, który chcemy złamać.

Tryby ataku (-a): Oprócz prostego ataku słownikowego, Hashcat oferuje inne tryby ataku, takie jak:

-a 1: Atak kombinacyjny

-a 3: Atak brute-force

-a 6: Hybrydowy atak słownikowy z maską

-a 7: Hybrydowy atak maski ze słownikiem

Każdy z tych trybów ma swoje zastosowania w zależności od scenariusza łamania haseł.

KunegundaKosek/ Hashcat



0
Contributors 0
Issues 0
Stars 0
Forks



KunegundaKosek/Hashcat

Contribute to KunegundaKosek/Hashcat development by creating an account on GitHub.

 GitHub