Kompresja tablic obliczeń wstępnych – alternatywa dla tęczowych tablic

Michał Trojnara

Michal.Trojnara@pl.abnamro.com

ABN·AMRO

Cel prezentacji

- Zaproponowanie rozwiązania alternatywnego wobec popularnych ataków na algorytmy skrótów kryptograficznych przy użyciu tęczowych tablic
- Porównanie własności konkurencyjnych rozwiązań

Plan prezentacji

- Historia rozwiązań
  - Atak wyczerpujący
  - Hellman (1980): Łańcuchy
  - Rivest (1982): Punkty rozróżnialne
  - Oechslin (2003): Tęczowe tablice
- Rozwiązanie autorskie: Kompresja tablic obliczeń wstępnych
- Porównanie efektywności rozwiązań

Problem odwracania skrótów

- Wiele aplikacji przechowuje skróty haseł nie dodając do nich losowej wartości utrudniającej korzystanie z obliczeń wstępnych (ang. salt)
- Odwracając skrót otrzymujemy hasło do systemu
- Wybrane przykłady podatnych systemów
  - Windows Lan Manager (duże litery, max. 7 znaków, DES)
  - Hasło konta SYSTEM w Oracle (duże litery, DES)
  - Windows NT hash (MD4)
  - Cisco PIX (MD5)
  - Wiele aplikacji webowych
  - Wiele formatów szyfrowania plików

Atak wyczerpujący

- Dla danego skrótu *h* sprawdzamy dopuszczalne hasła $p \in P$
- Zatrzymujemy po znalezieniu $H(p)=h$
- W systemach rozproszonych zakresy haseł można pobierać z centralnego serwera lub losować („chińska loteria”)

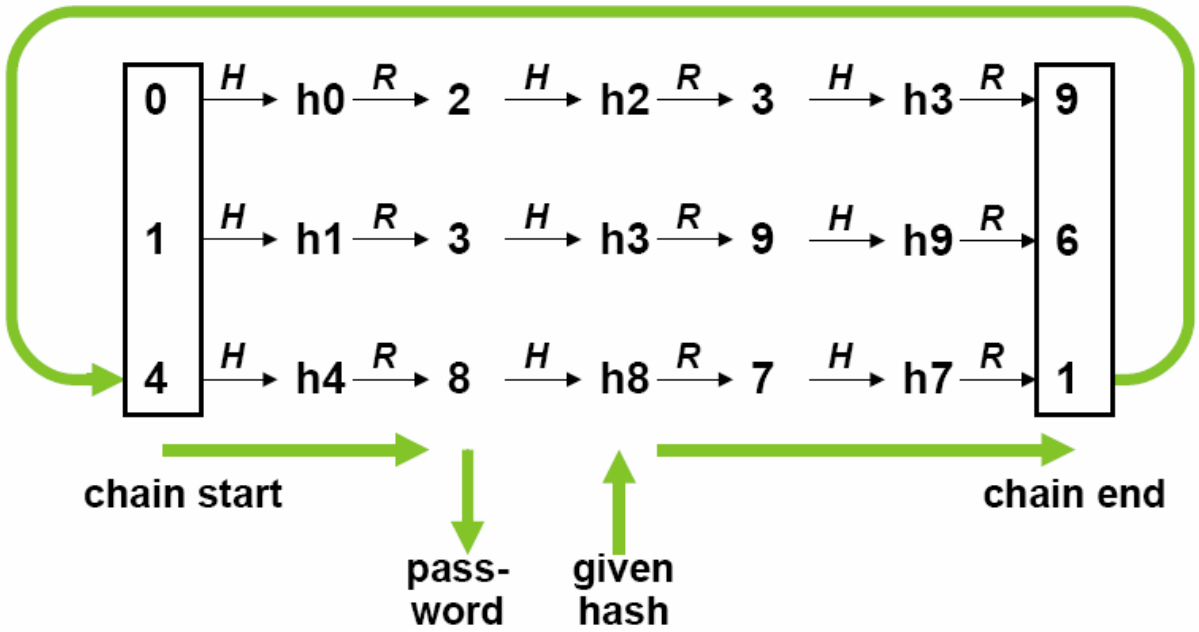
Chińska loteria (RFC 3607)

RFC 3607 – „Chinese Lottery Cryptanalysis Revisited: The Internet as a Codebreaking Tool”

- Estymata dla roku 2002 (*100,000,000* maszyn w Internecie)
- Założona wielkość botnetu: *503,968* maszyn
- DES
  - Zagregowana wydajność: *2.88e+11* kluczy/sekundę
  - Czas łamania: *1.45* dnia
  - Czas dla 8-znakowych haseł: *16.29* minut
- MD5
  - Zagregowana wydajność: *9.79e+11* kluczy/sekundę
  - Czas dla 64-bitowych kluczy: *218.04* dni
  - Czas dla 8-znakowych haseł: *4.79* minuty

Łańcuchy – opis algorytmu

W pamięci zapisywane są tylko pary początek/koniec łańcucha iteracji



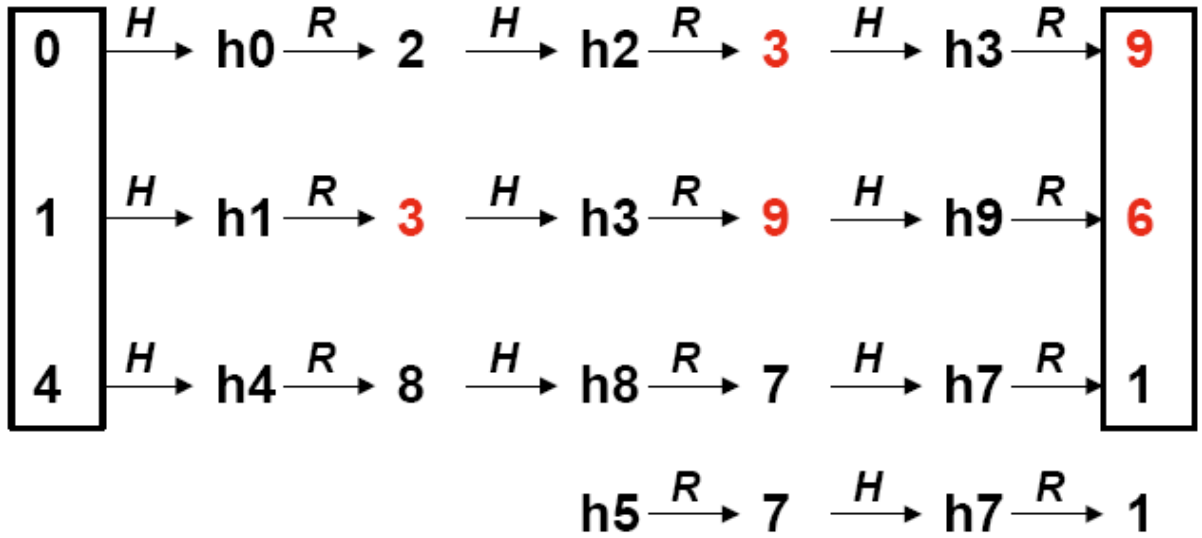
Philippe Oechslin, Objectif Sécurité „Rainbow Cracking: Do you need to fear the Rainbow?”

Łańcuchy – punkty rozróżnialne

- Problem z obliczaniem łańcuchów
  - Każda iteracja wymaga wyszukania w tablicy łańcuchów
- Zaproponowane rozwiązanie
  - Zamiast generować łańcuchy stałej długości można kończyć je na wartości spełniającej proste kryterium (np. *d* wybranych bitów ma wartość *0*)

Łańcuchy – scalanie

Funkcja redukcji może mieć tą samą wartość dla różnych skrótów
→ skutkiem mogą być fałszywe alarmy



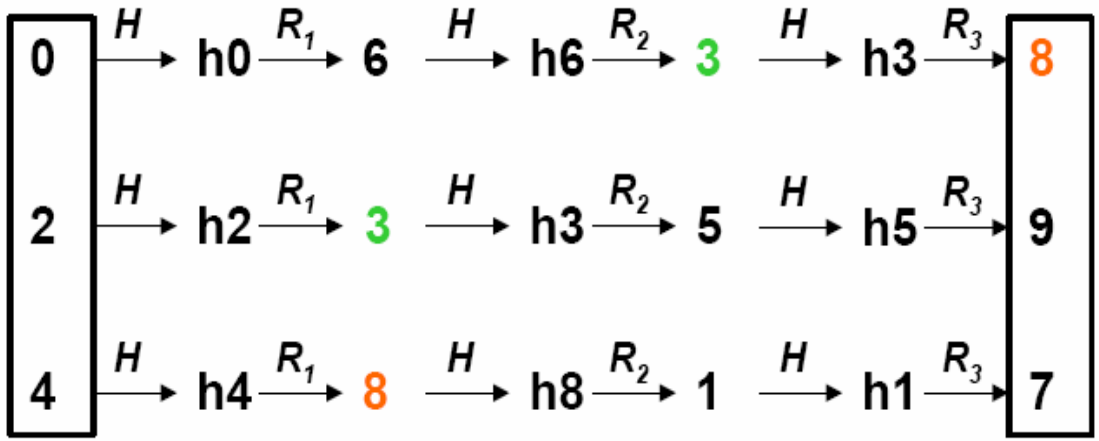
Philippe Oechslin, Objectif Sécurité „Rainbow Cracking: Do you need to fear the Rainbow?”

Łańcuchy – właściwości

- Wymagania dla *N* dopuszczalnych haseł:
  - $N^{2/3}$ par początek/koniec łańcucha w pamięci
  - $O(N^{2/3})$ obliczeń (iteracji funkcji skrótu)
  - $O(N)$ obliczeń wstępnych
- Prawdopodobieństwo sukcesu: *80%*

Tęczowe tablice – opis algorytmu

- Różne fukcje redukcji w kolejnych iteracjach
- Łańcuchy ulegają scaleniu tylko wtedy, kiedy to samo hasło występuje w obu łańcuchach na tej samej pozycji



Philippe Oechslin, Objectif Sécurité „Rainbow Cracking: Do you need to fear the Rainbow?”

Tęczowe tablice – właściwości

- Atak jest *20,000* razy szybszy niż atak wyczerpujący
- Przechowanie łańcucha wymaga w zależności od implementacji w przybliżeniu *32* bajty
- Średnio potrzebne jest *0.01* bajta na każdy skrót
- Dostęp do pamięci nie jest sekwencyjny – konieczność korzystania z pamięci o dostępnie swobodnym (RAM)
- Skuteczność (pokrycie przestrzeni haseł) zależy od wybranego punktu zakończenia obliczeń wstępnych → obliczenia wstępne są istotnie bardziej czasochłonne od pojedynczego ataku wyczerpującego

Kompresja tablic obliczeń wstępnych – opis algorytmu

- Algorytm generowania tablicy
  - Wygenerować wszystkie dopuszczalne hasła
  - Każde hasło zapisać po kompresji do pliku o numerze zależnym od części jego skrótu
  - Kompresja polega na zapisywaniu wyłącznie różnic pomiędzy hasłami w danym pliku
- Algorytm odwracania skrótu
  - Odczytać plik o numerze zależnym od skrótu
  - Obliczać skróty haseł zapisanych w pliku do znalezienia właściwego

Kompresja tablic obliczeń wstępnych – właściwości

- Atak jest *65,536* (2^{16}) razy szybszy niż atak wyczerpujący
- Do zapisania różnicy potrzeba średnio od *2* do *3* bajtów
- Dla 2^{37} dopuszczalnych haseł każdy plik ma co najwyżej $3*2^{37}/2^{16} = 3*2^{21} = 6$ MB
- Do znalezienia wartości wystarczy przeszukać jeden plik, co zajmie w przybliżeniu $2^{21}/10^6 = 2$ sekundy
- Dostęp do pamięci jest sekwencyjny – możliwość korzystania z pamięci masowej (HDD)
- Czas obliczeń wstępnych taki sam, jak czas pojedynczego ataku wyczerpującego

(obliczenia przeprowadzone dla sugerowanej liczby 2^{16} plików)

Porównanie algorytmów dla 2^{37} haseł LM DES

Algorytm	Tęczowe tablice	Tablice obliczeń wstępnych
Wykorzystanie pamięci	*1.4* GB	$3*2^{37}$ = *384* GB
Rodzaj pamięci	RAM	HDD
Cena 1GB	*400* PLN	*1* PLN
Koszt pamięci	*560* PLN	*384* PLN
Wydajność	*13.6* sekundy	~ *2* sekundy
Skuteczność	*99.9%*	*100%*
Czas obliczeń wstępnych	~ *300* godzin	~ *40* godzin

kolor niebieski - wartość empiryczna

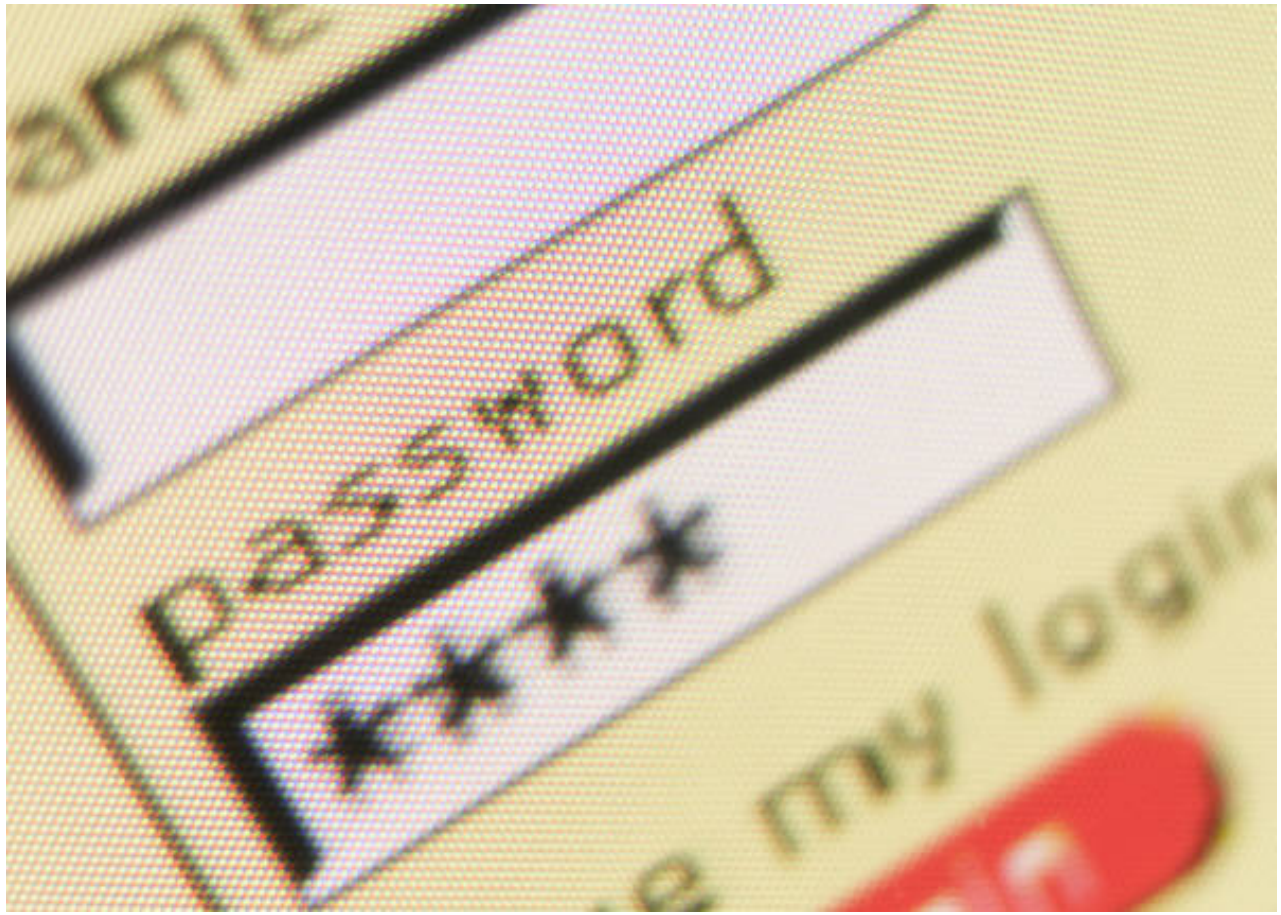
Wnioski

- Zaproponowane rozwiązanie ma pod wieloma względami lepsze własności niż tęczowe tablice
  - Niższy koszt pamięci
  - Lepsza wydajność odwracania skrótów
  - Lepsza wydajność obliczeń wstępnych
  - Większa skuteczność – możliwość odwrócenia *100%* skrótów danych należących do przestrzeni haseł
- Przewagą tęczowych tablic jest możliwość powielenia wyniku obliczeń wstępnych przez sieć – dla algorytmu kompresji tablic obliczeń wstępnych byłoby to niepraktyczne
- Niskie ceny twardych dysków pozwalają przechowywać tablice kluczy o długości do *40* bitów przy stosunkowo niewielkim koszcie

Literatura

- Philippe Oechslin „Making a Faster Cryptanalytic Time-Memory Trade-Off”
- Philippe Oechslin „Rainbow Cracking: Do you need to fear the Rainbow?”
- Jean-Jacques Quisquater, François-Xavier Standaert: „Exhaustive Key Search for DES: Updates and refinements”
- RFC 3607 - Chinese Lottery Cryptanalysis Revisited: The Internet as a Codebreaking Tool

Dziękuję za uwagę



Michal.Trojnara@pl.abnamro.com

ABN·AMRO