Strategi Pemilihan Kata Sandi Berdasarkan Serangan yang Berbasis Kombinatorial dan Probabilitas

Shevalda Gracielira 13516134
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13516134@std.stei.itb.ac.id

***Abstract*— Kata sandi pada penggunaan aplikasi maupun situs sudah cukup biasa. Namun, saran yang sering diberikan saat pengguna mendaftarkan akunnya adalah untuk memakai kata sandi dengan panjang tertentu maupun dengan setidaknya tiga jenis karakter yang biasanya adalah alfabet kecil, alfabet besar, dan angka. Kedua saran tersebut mungkin cukup efektif jika hanya mempertimbangkan serangan *brute force*. Namun, belajar dari mesin Enigma yang memang mempunyai banyak konfigurasi namun tetap dapat dipecahkan, begitu juga dengan cara-cara penyerangan di era internet. Penyerang sudah memakai algoritma-algoritma seperti *dictionary attack*, *mangled attack*, dan sejenisnya. Makalah ini akan melihat kenapa saran panjang karakter dan kombinasi jenis karakter adalah saran yang paling lumrah diberikan dan kenapa saran-saran tersebut tidak sepenuhnya efektif.**

***Keywords*—Kata sandi, mesin Enigma, *dictionary attack*, *mangled attack*.**

I. PENDAHULUAN

Penggunaan kata sandi masih merupakan cara yang paling sering dijumpai untuk membuktikan kepemilikan sebuah akun ataupun membuktikan seseorang yang mempunyai wewenang untuk mengakses suatu layanan. Sering kali kata sandi ini digunakan di situs atau aplikasi yang cukup sensitif seperti perbankan ataupun pemerintahan. Dengan kemungkinan kata sandi dapat diperoleh untuk mengeksploitasi sistem-sistem sensitif ini, maka pemilihan kata sandi yang susah untuk ditebak merupakan suatu hal yang vital pada era digital sekarang.

Saran yang biasa diberikan ketika membuat sebuah kata sandi di halaman pembuatan akun adalah untuk menggunakan kata sandi dengan suatu panjang minimum. Saran lain adalah menggunakan kombinasi jenis karakter yang berbeda untuk membuat kata sandi. Kombinasi karakter yang sering disarankan adalah angka, alfabet kecil, dan alfabet besar.

Saran-saran tersebut mungkin cukup efektif jika pengguna menggunakan kombinasi jenis karakter tersebut dalam urutan yang tidak berpola. Dengan pemilihan kata sandi yang masih berpola, saran ini masih memiliki kelemahan karena tidak mempertimbangkan penyerangan yang sudah menggunakan algoritma untuk memprediksi kemungkinan kombinasi dengan urutan tertentu.

Offline attack adalah serangan yang biasa mengeksploitasi pola ini ketika penyerang memperoleh kata sandi yang masih merupakan hasil *password hashing*. Terdapat beberapa *offline attack*, seperti *brute-force*, *dictionary attack*, dan *mangled dictionary attack*.

Beberapa serangan memperoleh kata sandi tanpa harus mencoba semua kemungkinan kata sandi. Salah satu caranya adalah menjebak pengguna untuk mengirimkan kata sandinya lewat halaman masuk yang palsu. Cara lain adalah menyusup pada jaringan sehingga dapat memperoleh kata sandi sebelum kata sandi tersebut dienkripsi oleh *password hashing*. Namun, jenis-jenis serangan seperti lebih mengarah pada cara mengakali atau merusak sistem. Serangan tersebut tidak mencakup faktor probabilitas dan kombinasi yang dimaksud dalam makalah ini. Serangan sejenis ini tidak akan dibahas lebih lanjut.

Sebelum mulai menganalisis serangan-serangan tersebut, ada baiknya juga melihat kriptografi pada Perang Dunia II, yaitu mesin Enigma. Jika dilihat dari banyaknya kemungkinan konfigurasi yang dapat dibangkitkan pada mesin Enigma, tentu tidak heran mengapa pihak Jerman sangat yakin bahwa mesin ini merupakan mesin enkripsi yang sempurna. Namun, dengan mempelajari kelemahan-kelemahan mesin Enigma dan bagaimana kelemahan ini dieksploitasi oleh pihak musuh, pengguna situs atau aplikasi dapat menyadari kelemahan sistem kata sandi dan membuat kata sandi yang lebih kuat sebagai tindakan preventif terhadap serangan-serangan yang berusaha memperoleh kata sandi.

II. LANDASAN TEORI

A. Kaidah Menghitung Kombinatorial

Dalam kombinatorial, terdapat dua teknik menghitung yaitu kaidah perkalian dan kaidah penjumlahan.

Kaidah perkalian adalah kaidah yang digunakan ketika menghitung berapa banyak kemungkinan untuk dua atau lebih percobaan terjadi pada saat yang bersamaan. Contohnya adalah seperti pemilihan acak dalam sebuah kelas berisi sepuluh mahasiswa dan tujuh mahasiswi. Jika dosen memilih satu mahasiswa dan satu mahasiswi, banyak kemungkinan pasangan yang dipilih adalah $10 \times 7 = 70$ pasangan.

Kaidah penjumlahan adalah kaidah yang digunakan ketika dua atau lebih percobaan dapat terjadi tetapi hanya satu percobaan yang dapat terjadi pada satu waktu. Mengambil contoh jumlah mahasiswa dan mahasiswi di atas, misalkan dosen memilih mahasiswa tanpa melihat jenis kelaminnya.

Banyak kemungkinan orang yang dipilih adalah 10 + 7 = 17 orang.

Karena kaidah menghitung kombinatorial ini dapat digunakan untuk dua atau lebih kejadian, maka kaidah ini dapat diperluas sebagai berikut:

a. $k_1 \times k_2 \times k_3 \times \ldots \times k_n$ untuk kaidah perkalian dengan *n* kejadian.
b. $k_1 + k_2 + k_3 + \ldots + k_n$ untuk kaidah pertambahan dengan *n* kejadian.

B. Permutasi

Permutasi adalah jumlah kemungkinan urutan yang dapat dibentuk dengan sejumlah pengaturan atau kejadian. Permutasi menggunakan kaidah perkalian dengan jumlah objek atau kejadian sebanyak *n*, maka dapat banyak kemungkinan urutan yang dapat terbentuk dapat dituliskan sebagai berikut

$$n \times (n - 1) \times (n - 2) \times ... \times 1 = n!$$

Jika permutasi tersebut diterapkan untuk *n* objek yang akan ditaruh pada *r* tempat atau wadah, maka permutasinya dapat ditulis sebagai berikut

$$P(n, r) = n(n - 1)(n - 2)\ldots(n - (r - 1))$$

$$P(n, r) = \frac{n!}{(n - r)!}$$

C. Probabilitas

Probabilitas, atau dikenal juga sebagai peluang, adalah teori tentang seberapa besar kemungkinan sebuah kejadian terjadi dengan acuan kejadian-kejadian yang lain. Ada dua istilah yang digunakan dalam probabilitas, yaitu ruang contoh (*sample space*) dan titik contoh (*sample point*).

Ruang contoh adalah kumpulan atau himpunan kejadian yang mungkin terjadi dalam hasil percobaan. Titik contoh adalah hasil-hasil yang ada di dalam kumpulan atau himpunan kejadian tersebut. Jika ruang contoh dapat dilambangkan sebagai *S* dan titik contoh sejumlah *n* dapat dilambangkan dengan x, maka hubungan keduanya dapat ditulis sebagai berikut

$$S = \{ x_1, x_2, x_3, \ldots, x_n\}$$

Peluang sebuah kejadian adalah peluang sebuah titik contoh dapat terjadi di dalam sebuah ruang contoh yang terbatas. Peluang kejadian *E* dalam ruang contoh *S* dapat ditulis sebagai berikut

$$p(E) = \frac{|E|}{|S|} = \sum_{x_i \in E} p(x_i)$$

III. Enkripsi dan Dekripsi Mesin Enigma

Kata sandi biasanya disimpan dalam bentuk enkripsi hasil *password hashing*. Sebelum menguraikan serangan-serangan yang berusaha memperoleh kata sandi walau masih dalam bentuk *chipertext*, ada baiknya pengguna situ ataupun aplikasi mengerti bagaimana mesin Enigma, yang dinyatakan sebagai mesin enkripsi yang mustahil didekripsi pada zamannya, akhirnya berhasil ditaklukkan juga.

Mesin Enigma adalah mesin enkripsi pihak Jerman pada Perang Dunia II. Mesin ini diciptakan karena komunikasi pada Perang Dunia II menggunakan radio yang dapat dengan mudah ditangkap oleh pihak musuh. Oleh karena alasan tersebut, pihak Jerman berusaha membuat sebuah mesin enkripsi supaya pihak musuh tidak tahu isi dari pesan yang dikirim jika berhasil menangkap pesan tersebut. Permasalahan ini dipecahkan oleh mesin Enigma.

Tampilan mesin Enigma mirip dengan mesin tik yang biasa digunakan pada zaman tersebut untuk membuat pesan telegram. Yang membedakan mesin Enigma dengan mesin tik umumnya adalah bagian lampu dengan label huruf yang akan menyala ketika sebuah huruf ditekan di mesin tik tersebut. Inilah sebagian dari rahasia cara kerja enkripsi mesin Enigma. Mesin Enigma memanfaatkan cara kerja mesin secara mekanik dan elektrik.

Dari segi mekanik, mesin Enigma mempunyai tiga buah rotor yang berputar untuk setiap kali sebuah huruf atau titik ditekan pada mesin. Setiap rotor mempunyai 26 konfigurasi yang berbeda untuk setiap hurufnya. Awalnya rotor paling kanan yang akan berputar untuk setiap huruf yang ditekan pada mesin Enigma. Jika 26 konfigurasi tersebut sudah digunakan, maka rotor kedua diputar oleh rotor pertama tadi dan seterusnya. Dengan konfigurasi tersebut, banyak konfigurasi yang dapat dibangkitkan dari tiga rotor tersebut adalah 26 × 26 × 26 = 17,576 konfigurasi.

Dari segi elektrik, mesin Enigma mempunyai papan steker (*plugboard*). Papan steker ini berfungsi untuk mensubstitusi sebuah huruf dengan huruf lain sebelum dan sesudah dienkripsi oleh tiga rotor di dalam mesin Enigma. Misalnya kabel huruf ‘A’ dipasang pada steker huruf ‘L’, maka jalur yang diikuti dalam mesin Enigma adalah jalur ‘L’ ketika huruf ‘A’ yang ditekan di mesin Enigma. Sebaliknya, jika huruf ditekan ‘L’ di mesin Enigma, maka jalur yang dibangkitkan adalah jalur huruf ‘A’. Pada jalur keluar, hal sama juga terjadi. Hasil akhir dari enkripsi tersebut ditandai oleh huruf yang menyala di bagian lampu mesin Enigma. Dengan konfigurasi kabel sebanyak enam buah, maka akan menghasilkan 100,391,791,500[1] pasangan huruf.

Gambar 1 Penampakan mesin *chiper* Enigma yang digunakan pihak Jerman pada Perang Dunia II.
Sumber: http://users.telenet.be/d.rijmenants/en/enigma.htm

Dengan begitu banyak konfigurasi, ternyata untuk mendekripsi pesan yang dihasilkan Enigma cukup sederhana. Operator yang akan mendekripsi pesan tersebut hanya membutuhkan posisi awal dari masing-masing rotor dan posisi kabel dipasang pada papan steker.

Melihat banyak jumlah konfigurasi tersebut, tentu saja pihak Jerman yakin bahwa enkripsi mereka tidak akan terpecahkan. Belum lagi pada versi berikutnya, pihak Jerman menambah jumlah rotor menjadi empat dan jumlah kabel di papan steker menjadi sepuluh. Konfigurasi tersebut bertambah menjadi 456,976 untuk empat rotor dan 150,738,274,937,250[2] untuk sepuluh kabel tersebut.

Namun, kenapa dengan begitu banyak konfigurasi yang hampir mustahil dipecahkan pada zaman tersebut, mesin Enigma masih bisa didekripsi oleh pihak musuh? Ternyata, kelemahan mesin Enigma terletak pada sistemnya sendiri dan penggunaannya.

Dari segi sistemnya, mesin Enigma mempunyai kelemahan berikut

1. Sebuah huruf tidak mungkin dienkripsi menjadi dirinya sendiri sehingga kemungkinan yang dihasilkan tidak sebanyak kemungkinan ketika sebuah huruf dapat menjadi dirinya sendiri.
2. Kabel yang mengubungkan satu huruf dengan huruf lain bersifat timbal-balik sehingga mengurangi konfigurasi yang harus dicoba oleh musuh dibanding jika kabel untuk jalur masuk dan jalur keluar dipisah.
3. Derajat perputaran untuk setiap rotor berbeda. Karena perbedaan derajat ini, urutan untuk tiap rotor menjadi tidak lebih unik dibanding perputaran derajat yang sama semua.

Dari segi penggunaannya, berikut permasalahan yang ditemukan

1. Pada masa uji coba mesin Enigma, terdapat beberapa *plaintext*, *chipertext*, dan kunci dari sandi yang diperoleh dari instruksi manual mesin Enigma tersebut.
2. Kunci untuk mesin Enigma yang mudah ditebak seperti AAA, BBB, dan tiga huruf berurutan secara horizontal maupun diagonal di mesin Enigma.
3. Pada beberapa jaringan, ada peraturan yang menyatakan bahwa tidak ada konfigurasi rotor yang sama jika telah dipakai sebelumnya dan dalam bulan yang sama. Karena pembatasan konfigurasi ini, pihak musuh dapat mengurangi konfigurasi yang harus dicoba.
4. Di sisi lain, beberapa konfigurasi dipakai ulang untuk bulan-bulan selanjutnya sehingga konfigurasi bulan sebelumnya dapat dicoba pada konfigurasi bulan ini.
5. Untuk penghubungan kabel di papan steker, ada beberapa operator yang mengharuskan kabel satu huruf tidak dipasang pada steker huruf yang bersebelahan.

Mesin Enigma mungkin saja menjadi mesin enkripsi yang sempurna untuk zamannya. Namun, dengan kelemahan dari sisi sistem dan penggunaannya, mesin Enigma akhirnya dapat ditaklukkan.

Dari pelajaran pemecahan ini, pengguna mesin enkripsi seharusnya lebih sadar sehingga dapat mencegah pihak musuh mengenkripsi pesan-pesan selanjutnya dalam perang tersebut.

IV. Mengenal Jenis Serangan untuk Kata Sandi

Dari enkripsi dan deskripsi mesin *chipper* Enigma, dapat dilihat bahwa dengan mengenali jenis serangan terhadap kata sandi, pengguna sebuah aplikasi atau situs dapat melakukan tindakan preventif supaya penyerang tidak mudah menggunakan kata sandi yang masih dienkripsi oleh *password hashing*.

Jenis serangan yang akan dibahas di sini adalah *dictionary attack*, *mangled attack*, *Markov-chain based attack*, dan *brute-force attack*.

Pada bagian ini, data yang digunakan diperoleh dari data penelitian *Password Strength: An Empirical Analysis*[8] yang mengolah data kata sandi dari tiga tempat berbeda, yaitu

- *The "Italian" dataset* (IT) di mana data kata sandi diperoleh dari sebuah aplikasi *instant messaging* di Italia. Tidak ada panduan kepada pengguna bagaimana membuat kata sandi sehingga cukup aman untuk mengasumsikan jika kata sandi yang dibuat tidak terlalu kuat dibanding kata sandi dari *dataset* lain.
- *The "Finnish" dataset* (FI) di mana kata sandi diperoleh dari sebuah situs forum Finlandia. Kata sandi yang digunakan adalah yang belum dienkripsi.
- *The MySpace dataset* (MS) di mana kata-kata sandi diperoleh setelah pengguna MySpace mencoba masuk ke akun mereka pada halaman masuk yang palsu. Kata sandi pada MySpace diharuskan menggunakan huruf alfabet dan non-alfabet.

Selain itu, data-data di bawah diperoleh juga dari *Password Pattern and Vulnerability Analysis for Web and Mobile Applications* yang mendasarkan data mereka pada data-data kata sandi dari akun situs yang telah dibocorkan. Situs-situs ini mencakup Rockyou[1] dan 163com[2].

A. *Dictionary attack*

Dictionary attack adalah serangan yang mengandalkan *database* yang berisi kemungkinan-kemungkinan kata sandi. Biasanya kata sandi yang lemah akan dengan cepat diserang oleh jenis serangan ini. Serangan ini cukup efektif karena salah satu ciri-ciri kata sandi yang lemah adalah kata sandi yang lumrah dipakai pengguna-pengguna lain karena mudah diingat, seperti "*password*" dan "*letmein*".

Salah satu program yang memanfaatkan *dictionary attack* adalah John the Ripper[3] (JtR) yang merupakan alat untuk memulihkan kata sandi. Berikut hasil dari analisis kata sandi menggunakan *dictionary attack*.

[1] https://rockyou.com/
[2] www.163.com/
[3] http://www.openwall.com/john/

Dictionary (size)	IT		FI	
	Found	Guess pr.	Found	Guess pr.
Frequent (2.8K)	5.95%	$2.1 \cdot 10^{-5}$	2.86%	$1.0 \cdot 10^{-5}$
English 1 lc (27K)	4.91%	$1.8 \cdot 10^{-6}$	3.38%	$1.2 \cdot 10^{-6}$
English 2 lc (297K)	9.42%	$3.2 \cdot 10^{-7}$	6.26%	$2.1 \cdot 10^{-7}$
English 3 lc (390K)	11.59%	$3.0 \cdot 10^{-7}$	7.53%	$1.9 \cdot 10^{-7}$
Extra lc (445K)	8.03%	$1.8 \cdot 10^{-7}$	8.16%	$1.8 \cdot 10^{-7}$
Italian 1 lc (63K)	3.71%	$5.9 \cdot 10^{-7}$	0.79%	$1.3 \cdot 10^{-7}$
Italian 2 lc (344K)	14.89%	$4.3 \cdot 10^{-7}$	6.62%	$1.9 \cdot 10^{-7}$
Finnish lc (359K)	8.45%	$2.4 \cdot 10^{-7}$	20.24%	$5.6 \cdot 10^{-7}$
All above (1.45M)	24.79%	$1.7 \cdot 10^{-7}$	26.02%	$1.8 \cdot 10^{-7}$
All JtR dicts (3.9M)	25.94%	$6.6 \cdot 10^{-8}$	26.97%	$6.6 \cdot 10^{-8}$
Mnemonics (406K)	1.27%	$3.1 \cdot 10^{-8}$	0.35%	$8.7 \cdot 10^{-9}$

Tabel 1 Data yang berdasarkan *dictionary attack* beserta persentasenya. Sumber [8]

Kamus English 1, English 2, dan English 3 menyatakan ukuran kamus tersebut. English 3 adalah kamus dengan ukuran terbesar dan mencakup kamus English 1 dan English 2. Begitu juga dengan Italian 1 dan Italian 2. "lc" menandakan kata sandi yang menggunakan jenis huruf kecil semua. Data MS tidak ada pengolahan data ini karena keharusan pemilik kata sandi untuk menggunakan jenis karakter alfabet dan non-alfabet.

Probabilitas tebakan (*guess probability*) yang ditampilkan pada data menunjukkan titik contoh yang merupakan kata sandi yang ada di dalam kamus dibandingkan dengan ruang contoh kamus itu sendiri.

Jika dianalisis dari data tersebut, *dictionary attack* hanya efektif sampai ukuran tertentu saja. Dilihat dari perbandingan probabilitas tebakan kamus English 1 dengan probabilitas tebakan kamus English 2, probabilitas tebakan tidak bertambah melainkan berkurang. Hal ini dimungkinkan karena tidak adanya pertambahan pada titik contoh ketika ruang contoh bertambah.

Dari data tersebut, dapat dilihat bahwa *dictionary attack* hanya efektif jika kata sandi mengandung satu kata dan merupakan kata yang terdapat pada *database* kamus kata sandi yang sering digunakan. Penggunaan kata dari bahasa selain bahasa Inggris dapat mengurangi probabilitas tebakan.

Probabilitas tebakan tersebut dapat ditingkatkan jika data-data personal pemilik kata sandi juga ditambahkan dalam *database* kamus seperti tanggal lahir, alamat rumah, saudara, atau hal-hal personal lainnya.

B. *Mangled attack*

Mangled attack sebenarnya cukup mirip dengan *dictionary attack*. Perbedaannya adalah *mangled dictionary attack* memanfaatkan potongan dari kata sandi dari *database* kamus dan menggabungkannya dengan potongan lain yang memungkinkan.

Serangan jenis ini lebih menguntungkan karena tidak hanya mengandalkan satu jenis karakter untuk satu kata sandi tetapi bisa merupakan gabungan dari jenis-jenis karakter yang berbeda. Keuntungan ini mungkin dapat bekerja jika pemilik kata sandi menggunakan kata yang sebenarnya umum namun hanya diganti beberapa huruf dengan angka yang mirip dengan huruf tersebut seperti menggantikan alfabet 'a' atau 'A' dengan angka 4. Salah satu teknik khusus dalam *mangled attack* adalah *probabilistic context-free grammar*.

Probabilistic context-free grammar memanfaatkan pola jenis karakter yang digunakan pada sebuah kata sandi. Kata sandi yang menggunakan beberapa jenis karakter kemungkinan menggunakan dengan pola tersebut. Hal ini yang membuat *probabilistic context-free grammar* cukup efektif. Untuk meningkatkan tingkat keefektifan serangan, penyerangan hanya perlu mempertimbangkan beberapa jenis karakter yang sering digunakan untuk membuat sebuah kata sandi.

Character-set	Percentage (%)	Number of items
Numeric	58	2,931,867
loweralphanum	30	1,527,719
loweralpha	08	450,746
loweralphaspecialnum	00	38,913
mixedalphanum	00	26,097
upperalphanum	00	23,905
specialnum	00	15,614
loweralphaspecial	00	4830
mixedalpha	00	4353
upperalpha	00	3142
All	00	2172
upperalphaspecialnum	00	1722
mixedalphaspecial	00	550
Special	00	164
upperalphaspecial	00	133

Tabel 3 Persentase penggunaan jenis karakter pada kata sandi di situs 163com. Sumber [9]

Character-set	Percentage (%)	Number of items
loweralphanum	88	4,720,183
upperalphanum	06	325,942
mixedalphanum	05	293,432

Tabel 3 Persentase penggunaan jenis karakter pada kata sandi di situs Rockyou. Sumber [9]

Kedua tabel data tersebut memunculkan sebuah hipotesis bahwa jenis karakter yang paling sering digunakan adalah angka, alfabet kecil, dan alfabet besar. Dengan menggunakan data di atas, penyerang tidak perlu memperhitungkan semua kemungkinan jenis karakter untuk menebak satu karakter dalam kata sandi. Kombinasi angka, alfabet kecil, dan alfabet besar sudah cukup untuk memecahkan sebagian besar kata sandi.

Berikut data keberhasilan penyerang jika serangan berdasarkan *probabilistic context-free grammar* dijalankan.

Expression	Example	IT	FI	MS
[a-z]+	abcdef	51.21%	53.06%	1.09%
[A-Z]+	ABCDEF	0.29%	0.17%	0%
[A-Za-z]+	AbCdEf	53.74%	54.04%	1.09%
[0-9]+	123456	9.10%	3.43%	0.15%
[a-zA-Z0-9]+	A1b2C3	93.43%	95.43%	90.43%
[a-z]+[0-9]+	abc123	14.51%	27.10%	77.39%
[a-z]+1	abcde1	0.26%	1.43%	19.89%
[a-zA-Z]+[0-9]+	aBc123	16.30%	28.03%	77.48%
[0-9]+[a-zA-Z]+	123aBc	1.80%	2.16%	5.76%
[0-9]+[a-z]+	123abc	1.65%	2.09%	5.75%

Tabel 4 Persentase dari kata sandi yang cocok dengan beberapa *regular expression*. Sumber [8]

Data di atas kurang lebih memverifikasi hipotesis sebelumnya di mana kebanyakan kata sandi, jika tidak menggunakan alfabet kecil saja, menggunakan kombinasi alfabet kecil, alfabet besar, dan angka.

Kebijakan MySpace untuk memaksakan penggunanya memiliki kata sandi yang terdiri dari huruf alfabet dan non-alfabet cukup menyumbangkan kontribusi terhadap persentase penggunaan huruf alfabet kecil saja yang sedikit. Namun, di sisi lain, karena pemaksaan ini, pemilik kata sandi lebih condong untuk sekedar menambah angka 1 di akhir kata sandi mereka. Hal ini dapat diamati dari baris [a – z] + 1. Hanya 0,26% pemilik kata sandi di TI dan 1,43% pemilik kata sandi di FI yang menambahkan angka 1 di akhir kata sandi mereka. Namun, terdapat 19,89% pemilik kata sandi di MS yang menambahkan angka 1 di akhir kata sandinya.

Probabilistic context-free grammar dapat meningkatkan keefektifan *mangled attack* yang juga memanfaatkan kamus dari *dictionary attack*. Keefektifan ini dapat diuraikan dalam kombinatorial dan probabilitas.

Misalkan seorang pengguna mempunyai kata sandi sepanjang enam karakter. Jenis karakter yang diperhitungkan hanya angka, huruf alfabet kecil, dan huruf alfabet besar. Jika pola dari hasil *probabilistic context-free grammar* tidak diketahui sehingga penyerang harus menebak semua kemungkinan yang dapat dibuat oleh angka, huruf alfabet kecil, dan huruf alfabet besar, banyak tebakan penyerang dalam kasus terburuk adalah

$$(26 + 26 + 10)^6 = 56{,}800{,}235{,}584 \text{ tebakan}$$

Probabilitas menebak kata sandi tersebut adalah 1.761×10^{-11}. Namun jika penyerang memperhitungkan *probabilistic context-free grammar* pada tebakannya, maka hasilnya menjadi seperti berikut

1. [a – zA – Z] [0 – 9] +
   $((26 + 26) + 10)^3 = 238{,}328$ tebakan
   Probabilitas tebakan = 4.196×10^{-6}

2. [a – z] + [0 – 9] +
   $26^3 + 10^3 = 18{,}576$ tebakan
   Probabilitas tebakan = 5.383×10^{-5}

3. [a – z] + 1
   $26^5 + 1 = 11{,}881{,}377$ tebakan
   Probabilitas tebakan = 8.417×10^{-8}

4. [a – zA – Z] + [0 – 9] +
   $(26 + 26)^3 + 10^3 = 141{,}608$ tebakan
   Probabilitas tebakan = 7.062×10^{-6}

5. [0 – 9] + [a – zA – Z]
   $10^3 + (26 + 26)^3 = 141{,}608$ tebakan
   Probabilitas tebakan = 7.062×10^{-6}

6. [0 – 9] + [a – z] +
   $10^3 + 26^3 = 18{,}576$ tebakan
   Probabilitas tebakan = 5.383×10^{-5}

Berdasarkan jumlah tebakan dan probabilitas tebakan di atas, dapat disimpulkan bahwa semakin sedikit jenis karakter yang dipakai dalam sebuah pola, maka semakin besar pula probabilitas tebakan. Kata terakhir yang pasti, seperti mengakhiri kata sandi dengan angka 1, dapat menambah probabilitas tebakan secara drastis. Pertambahan tersebut belum mempertimbangkan jenis huruf alfabet yang digunakan. Jika kata sandi hanya memakai huruf alfabet kecil, maka probabilitas tebakan meningkat drastis.

Probabilistic context-free grammar ini sangat cocok jika penyerang sudah mengetahui batasan yang diberikan oleh sebuah situs mengenai jenis karakter yang diwajibkan. Jika dilihat dari persentase setiap *dataset*, ternyata [a – zA – Z] [0 – 9] + adalah pola yang paling sering dijumpai di kata sandi dengan maupun tanpa batasan ketika pengguna situs membuat sebuah kata sandi. Hal tersebut dapat dilihat dari jumlah pemilik kata sandi dengan pola tersebut lebih dari 90% di setiap situs. Dengan begitu, penyerang kemungkinan besar mencoba pola tersebut dahulu dalam *mangled attack*-nya sebelum mencoba pola yang lain.

C. *Brute-force attack*

Pada awal makalah, telah disebutkan bahwa serangan *brute-force* kemungkinan besar yang memunculkan saran untuk menggunakan beberapa jenis karakter pada kata sandi dan panjang kata sandi yang melebihi suatu batasan yang diberikan oleh aplikasi atau situs.

Sebagai perbandingan, berikut dijabarkan banyak jumlah tebakan dan probabilitas tebakan untuk kata sandi yang

a. hanya menggunakan satu jenis karakter (huruf alfabet kecil)
b. menggunakan dua jenis karakter (huruf alfabet kecil dan huruf alfabet besar)
c. menggunakan tiga jenis karakter (angka, huruf alfabet kecil, dan huruf alfabet besar

dari panjang enam sampai delapan karakter.

1. Kata sandi dengan enam karakter
   a. Satu jenis karakter
      $26^6 = 308{,}915{,}776$ tebakan
      Probabilitas tebakan = 8.417×10^{-8}

   b. Dua jenis karakter
      $(26 + 26)^6 = 19{,}770{,}609{,}664$ tebakan
      Probabilitas tebakan = 5.058×10^{-11}

   c. Tiga jenis karakter
      $(26 + 26 + 10)^6 = 56{,}800{,}235{,}584$ tebakan
      Probabilitas tebakan = 1.761×10^{-11}

2. Kata sandi dengan tujuh karakter
   a. Satu jenis karakter
      $26^7 = 8{,}031{,}810{,}176$ tebakan
      Probabilitas tebakan = 1.245×10^{-10}

   b. Dua jenis karakter
      $(26 + 26)^7 = 1{,}028{,}071{,}702{,}528$ tebakan
      Probabilitas tebakan = 9.727×10^{-13}

c. Tiga jenis karakter
$(26 + 26 + 10)^7 = 3{,}521{,}614{,}606{,}208$ tebakan
Probabilitas tebakan = 2.840×10^{-13}

3. Kata sandi dengan delapan karakter
a. Satu jenis karakter
$26^8 = 208{,}827{,}064{,}576$ tebakan
Probabilitas tebakan = 4.789×10^{-12}

b. Dua jenis karakter
$(26 + 26)^8 = 53{,}459{,}728{,}531{,}456$ tebakan
Probabilitas tebakan = 1.871×10^{-14}

c. Tiga jenis karakter
$(26 + 26 + 10)^8 = 218{,}340{,}105{,}584{,}896$ tebakan
Probabilitas tebakan = 4.580×10^{-15}

Dari ketiga perbandingan tersebut, terlihat jelas bahwa penggunaan jenis karakter yang lebih variatif dan pertambahan panjang kata dapat mengkali-lipatkan jumlah tebakan yang harus dicoba oleh penyerang.

Walau cara serangan *brute-force* membutuh usaha lebih untuk menebak sebuah kata sandi, namun serangan jenis ini cukup efektif untuk kata sandi yang lebih pendek dan tidak mempunyai pola tertentu yang mungkin saja terlewatkan oleh *dictionary attack* maupun *mangled attack*.

V. Pemilihan Kata Sandi yang Kuat

Dari jenis serangan yang telah dipaparkan sebelumnya, pengguna dapat menyusun strategi dalam memilih kata sandi yang probabilitas tebakannya paling kecil dan membutuhkan usaha lebih untuk ditebak.

Menggunakan kata sandi yang panjang. Walau kata sandi yang panjang tidak sepenuhnya menjamin kekuatan kata sandi tersebut, namun setidaknya dapat mengurangi probabilitas tebakan untuk penyerang. Jika kata sandi terlalu panjang untuk ditebak, penyerang biasanya akan langsung menyerah karena usaha untuk menebak kata sandi tersebut tidak sebanding dengan hasil ketika memperoleh kata sandi tersebut.

Selalu menggunakan kombinasi huruf alfabet kecil, huruf alfabet besar, angka, dan huruf non-alfanumerik. Hal ini berdasarkan jenis karakter yang sering digunakan hanya huruf alfabet kecil, huruf alfabet besar, dan angka. Dengan memastikan setidaknya ada empat jenis karakter, termasuk jenis karakter yang jarang digunakan untuk kata sandi, penyerang akan lebih susah menebak kata sandi dengan ketiga cara serangan yang telah dibahas.

Jangan menggunakan suatu pola berulang-ulang dalam kata sandi. Salah satu yang membuat *probabilistic context-free grammar* di *mangled attack* efektif adalah dengan mengandalkan pengguna situs atau aplikasi menggunakan kata sandi yang mempunya pola tertentu yang diulang berkali-kali. Dengan membuat sebuah kata sandi yang acak, *mangled attack* tidak bisa menebak kata sandi tersebut. Untuk memastikan suatu kata sandi acak, pengguna situs atau aplikasi dapat menggunakan beberapa pola tanpa pengulangan seperti *ay1Ft22RFc3*. Kata sandi tersebut tidak akan ditemukan polanya karena memang sengaja menggunakan pola-pola yang berbeda.

Jangan menggunakan suatu kata utuh ataupun sebagian sebagai kata sandi. Terutama menggunakan kata dalam bahasa Inggris untuk dijadikan kata sandi karena kamus yang digunakan di *dictionary attack* mempunyai ruang contoh yang banyak untuk kata dalam bahasa Inggris. Menggunakan bahasa lokal yang sangat jarang digunakan mungkin dapat menambah kekuatan kata sandi. Namun, pilihan ini cukup riskan jika penyerang tahu latar belakang pemilik kata sandi. Kata sandi lebih baik tidak berupa suatu kata utuh maupun penggabungan dengan kata lain.

Untuk membantu pengguna memilih kata sandi yang cukup kuat, saran lain yang biasa diberikan adalah menggunakan mnemonik sebagai kata sandi. Mnemonik dapat dibuat dengan memilih suatu kalimat yang mudah diingat dan mengambil huruf pertamanya dari setiap katanya. Beberapa huruf tersebut dapat diubah menjadi huruf alfabet besar, angka, maupun simbol-simbol lain. Cara ini cukup efektif jika kalimat yang dipilih bukanlah kalimat dari kutipan terkenal dan hasil dari mnemonik ini tidak membentuk suatu pola yang dapat dikenali.

VI. Kesimpulan

Sebelumnya, kekuatan kata sandi kurang lebih diukur, jika tidak mempertimbangkan cara yang mengakali atau merusak sistem, dari panjang kata sandi tersebut dan kombinasi penggunaan huruf alfabet kecil, huruf alfabet besar, dan angka. Namun, jika mempertimbangkan jenis-jenis serangan untuk menebak kata sandi seperti *dictionary attack, mangled attack,* dan *brute-force attack*, maka dua saran tersebut masih belum cukup.

Setelah melihat cara kerja masing-masing serangan, dapat disimpulkan bahwa kata sandi yang kuat adalah kata sandi yang cukup panjang, menggunakan kombinasi setidaknya empat jenis karakter, tidak mempunyai pola tertentu, dan tidak menggunakan sebuah kata utuh ataupun sebagian dari sebuah bahasa, terutama bahasa Inggris.

VI. Ucapan Terima Kasih

Pertama-tama, saya mengucapkan terima kasih kepada Tuhan Yang Maha Esa. Oleh karena-Nya, pengerjaan makalah ini lancar sampai selesai. Saya ingin berterima kasih kepada Dra. Harlili M.Sc, Dr. Judhi Santoso M.Sc, dan Dr. Ir. Rinaldi Munir, MT selaku dosen mata kuliah Matematika Diskrit. Selain itu, saya berterima kasih juga kepada teman-teman yang telah memberikan saya semangat. Tidak lupa juga saya mengucapkan terima kasih kepada kedua orang tua saya atas dukungan mereka.

Referensi

[1] Singh, Simon (1999), *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography*, London: Fourth Estate, ISBN 1-85702-879-1
[2] Sale, T. (n.d.). *Military Use of the Enigma*. Retrieved December 02, 2017, from http://www.codesandciphers.org.uk/enigma/enigma3.htm

[3] Rejewski, Marian (1984c), *Summary of Our Methods for Reconstructing ENIGMA and Reconstructing Daily Keys, and of German Efforts to Frustrate Those Methods*: Appendix C of Kozaczuk 1984.
[4] Munir, R., 2010. *Matematika Diskrit* 4th ed., Bandung: INFORMATIKA.
[5] Dade, L. (n.d.). *How Enigma Machines Work*. Retrieved December 02, 2017, from http://enigma.louisedade.co.uk/howitworks.html
[6] Hern, A. (2014, November 14). *How did the Enigma machine work?* Retrieved December 01, 2017, from https://www.theguardian.com/technology/2014/nov/14/how-did-enigma-machine-work-imitation-game
[7] *Password Cracking*. web.cs.du.edu/~mitchell/forensics/information/pass_crack.html. Accessed 2 Dec. 2017.
[8] Amico, M.D., Michiardi, P. & Roudier, Y., *Password Strength: An Empirical Analysis*. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.175.5282&rep=rep1&type=pdf [Accessed December 1, 2017].
[9] Shancang, L.I., Romdhani, I. & Buchanan, W., 2016. *Password Pattern and Vulnerability Analysis for Web and Mobile Applications*. ZTE Communications, vol. 14. Available at: https://www.napier.ac.uk/~/media/worktribe/output-367587/password-pattern-and-vulnerability-analysis-for-web-and-mobile-applications.pdf [Accessed December 1, 2017].

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 3 Desember 2017

Shevalda Gracielira 13516134