STUDI MENGENAI *SIDE CHANNEL ATTACK*: *ACOUSTIC CRYPTANALYSIS*

Victor – NIM : 13503001

Program Studi Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : if13001@students.if.itb.ac.id

Abstrak

Dalam perkembangan dunia kriptografi, faktor kemanan menjadi masalah utama yang harus diperhatikan. Algoritma dibuat serumit mungkin agar penyerang tak dapat membongkar isi dari, misalnya, pesan penting yang telah terenkripsi. Namun serumit apapun sebuah algoritma *chiper*, akan menjadi sia-sia jika si penyerang dapat memperoleh kunci yang dibutuhkan untuk membuka sebuah *chipertext*. Ada banyak cara bagi seorang penyerang untuk dapat memperoleh kunci ini. Cara yang paling sederhana dan paling kasar adalah dengan menggunakan serangan fisik terhadap pemegang kunci agar "mau" memberikan kunci. Namun seiring dengan perkembangan teknologi, serangan pun dapat menjadi lebih cerdas dan terselubung. Teknologi yang ada saat ini mengizinkan penyerang untuk dapat mengetahui kunci yang dimasukkan pengguna ketika sedang diketik – tanpa pengguna mengetahui bahwa kuncinya telah dicuri orang lain.

Makalah ini membahas mengenai pencurian kunci itu dengan salah satu *side channel attack* yang mengeksploitasi suara, baik yang terdengar oleh telinga manusia ataupun tidak, yang dihasilkan oleh operasi komputasi atau operasi masukkan-keluaran (*input-output*). Pada tahun 2004, Dmitri Asonov dan Rakesh Agrawal dari IBM Almaden Research Center mengumumkan bahwa papan kunci komputer dan set tombol nomor yang ada pada telepon atau mesin ATM amat rentan terhadap serangan berdasarkan pada perbedaan suara yang dihasilkan oleh masing-masing tombol. Dengan menggunakan *neural network* atau beberapa model khusus – seperti Hidden Markov Model – tombol tertentu yang sedang ditekan dapat dengan mudah dikenal. Dengan menganalisis suara yang telah direkam, dapat diperoleh kembali text yang telah dimasukkan melalui tombol-tombol pada papan kunci atau set tombol nomor. Teknik ini mengizinkan seorang penyerang menggunakan alat perekam dengar untuk dapat memperoleh kata sandi, kata kunci, PIN, atau informasi keamanan lainnya.

Dalam makalah ini juga akan dibahas mengenai percobaan yang dilakukan mengenai serangan, mengapa papan kunci memiliki bunyi yang berbeda untuk tiap tombolnya, dan juga kemungkinan-kemungkinan cara pertahanan yang bisa dilakukan untuk menangkal serangan ini.

Kata kunci: acoustic cryptanalysis, Hidden Markov Model, neural network, side channel attack.

1. Pendahuluan

Radiasi yang dihasilkan oleh alat-alat elektronik telah lama menjadi topik hangat dalam bidang keamanan dan kerahasiaan data [7]. Radiasi elektromagnetik dan radiasi optik telah banyak digunakan sebagai sumber serangan. Sebagai contoh, Kuhn telah berhasil memperoleh tampilan dari sebuah monitor CRT dengan menggunakan radiasi pantulan optik tak langsung (*indirectly reflected optical emations*) [10]. Tak lama kemudian Kuhn juga berhasil melakukan serangan terhadap monitor LCD [11]. Radiasi akustik adalah salah satu sumber data untuk serangan. Para peneliti telah membuktikan bahwa radiasi akustik dari sebuah mesin cetak *matrix* membawa informasi penting atas teks yang dicetaknya [7]. Beberapa peneliti kemudian mengemukakan kemungkinan untuk memperoleh operasi-operasi CPU berdasarkan radiasi akustik tersebut. Belum lama ini, Dmitri Asonov dan Rakesh Agrawal menunjukkan bahwa adalah mungkin untuk memperoleh kembali teks dari radiasi akustik yang dihasilkan dari bunyi ketikan papan kunci.

Seperti dituliskan di atas, Asonov dan Agrawal menyelidiki radiasi akustik dari sebuah papan kunci komputer, atau lebih terperinci bunyi 'klik'-nya, untuk men'dengar'kan kata-kata apa saja yang diketikkan. Serangan ini berdasarkan hipotesis bahwa suara 'klik' berbeda – mungkin hanya ada sedikit beda – dari satu tombol ke tombol yang lain,

walaupun tampaknya bunyi ‘klik’ di semua tombol papan kunci terdengar sama di telinga manusia. Percobaan mereka menunjukkan bahwa sebuah *neural network* dapat dilatih untuk membedakan tombol papan kunci mana yang ditekan.

Serangan ini murah dan tidak merusak. Murah karena selain sebuah perangkat komputer, perangkat keras yang diperlukan hanya sebuah *parabolic microphone*. Tidak merusak karena serangan ini tidak memerlukan intrusi fisik ke dalam sistem; suara ‘klik’ dapat direkam pada jarak jauh yang bersesuaian dengan perangkat keras yang tersedia. Dalam studi ini juga akan dicari cara penanggulangan jenis serangan ini.

Sebagai tambahan, serangan ini juga dipelajari pada komputer *notebook*, telepon dengan set tombol nomor yang bersuara, dan set tombol nomor ATM. Latar belakang dari penyelidikan ini adalah adanya kemungkinan bahwa apa yang diketikkan pada alat-alat di atas bisa juga diperoleh dengan menggunakan serangan terhadap bunyi ‘klik’.

2. Metode Serangan

Serangan ini berdasarkan hipotesis bahwa suara ‘klik’ dapat berbeda – mungkin hanya ada sedikit beda – dari satu tombol ke tombol yang lain, walaupun bunyi ‘klik’ di semua tombol papan kunci terdengar sama di telinga manusia. Untuk mengklasifikasikan beda bunyi ‘klik’ ini digunakan *neural network*. Alasan mengapa dipilih *neural network* adalah karena sebelumnya *neural network* berhasil digunakan untuk menyelesaikan persoalan serupa, seperti proses identifikasi speker [18].

2.1. Persiapan Percobaan

Pertama perlu dispesifikasikan peralatan dan perangkat lunak yang digunakan dalam percobaan Asonov dan Agrawal.

Papan kunci. Digunakan beberapa tipe papan kunci. Kebanyakan tipe papan kunci yang digunakan untuk percobaan adalah sebuah papan kunci IBM dengan S/N 0953260, P/N 32P5100. Percobaan dengan lebih dari satu papan kunci menggunakan tiga papan kunci GE Power HO97798. Untuk percobaan dengan telepon menggunakan Siemens RP240 (M/N 62001).

Microphones. Digunakan sebuah *microphone* untuk PC sederhana dengan jarak pendek sekitar 1 meter dan sebuah *parabolic microphone* untuk ‘mencuri dengar’ dari jauh.

Computer omnidirection microphone: S/N 33-3026 yang dimanufaktur oleh RadioShack; respon frekuensi: 30Hz-15kHz; *impedance*: 1000 ohms +/-30%; sensitivitas: -68dB +/-3dB; tegangan operasi: 1.0 sampai 10 VDC.
Parabolic microphone: ‘Bionic Booster’ yang dimanufaktur oleh Silver Creek Industries; respon frekuensi: 30Hz-15kHz (-3dB *response*); *gain amp. cut off* at 90 dB; *overall system gain*: 40dB; sensitivitas: -46dB (0dB = 1 V/Pa).

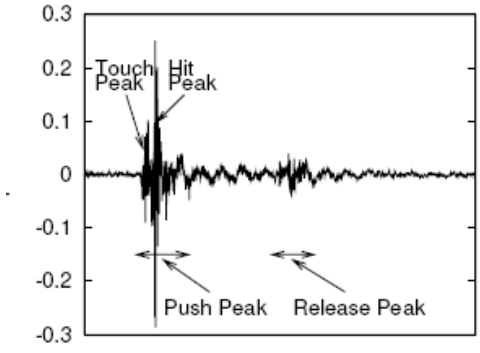
ADC dan FFT. *Input* masukkan suara kemudian di-*digital*-isasi dengan menggunakan *sound card* PC standar dengan 44.1 kHz *sampling rate*. Perangkat lunak Sigview v.1.81 digunakan untuk merekam suara sekaligus untuk menghitung *time*-FFT pada jendela 2 *ms*, dengan mengaplikasikan fungsi *windowing* Hanning.

Neural Network. Digunakan JavaNNS *neural network simulator* [23] untuk membangun sebuah *backpropagation neural network*. Jumlah *node* masukkan sama dengan ukuran fitur. Sebagai contoh, satu nilai per 20 Hz pada FFT memerlukan 200 *node* masukkan untuk interval 0-4 kHz. Akan ada 6-10 *node* tersembunyi (*hidden node*), tergantung dari ukuran fitur dan jumlah tombol. Jumlah *node* keluaran sama dengan jumlah tombol dalam percobaan. Satu *node* keluaran dihasilkan ketika dilakukan percobaan dengan dua tombol.

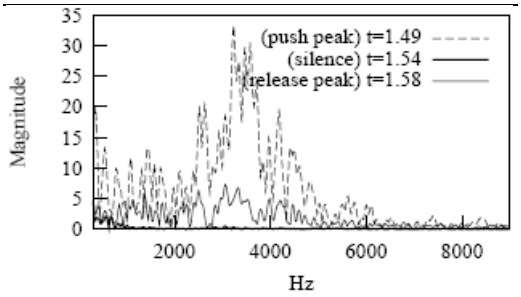
2.2. Melatih *Neural Network*

Suara mentah yang dihasilkan oleh bunyi ‘klik’ dari sebuah tombol tidak cukup bagus untuk dijadikan *training set* bagi *neural network*. *Neural network* direkomendasikan untuk dilatih menggunakan masukkan bernilai antara 0 dan 1 [19], yang berhubungan dengan masukkan sebesar kurang lebih 1 kB. Di lain pihak, ukuran dari sinyal akustik yang berhubungan dengan sebuah ketikan papan kunci adalah sebesar 10 kB. Oleh karena itu perlu dilakukan ekstraksi fitur-fitur yang relevan dari suara mentah tersebut.

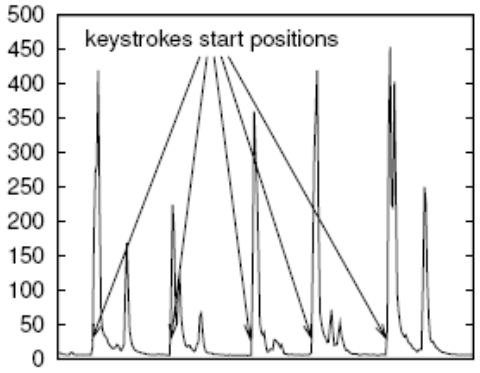
Diperlukan fitur-fitur yang mengizinkan *neural network* untuk dapat membedakan antar satu suara dengan suara yang lain dari ruang contoh suara. Frekuensi spektrum langsung diketahui memiliki variasi yang signifikan terhadap suara-suara yang terdengar mirip [8], yang membuatnya amat penting untuk percobaan ini. Yang menarik dari hal ini, frekuensi spektrum langsung ini juga digunakan sebagai fitur untuk pengklasifikasian suara yang umum [8].



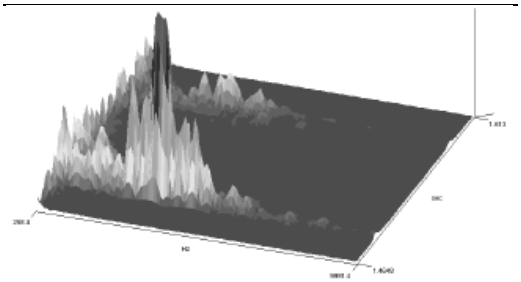
Gambar 1. Sinyal akustik dari satu 'klik'.



Gambar 2. Frekuensi spektrum berhubungan dengan puncak penekakan tombol, interval jeda, dan puncak pelepasan tombol.



Gambar 3. Ragam energi dari durasi selama 5 ketikan.



Gambar 4. *Time FFT* dari sinyal pada Gambar 1.

Juga diperlukan ketelitian untuk memilih waktu yang tepat ketika spektrum akan dikalkulasi. Untuk tujuan ini, sebuah pemahaman akan bagaimana sebuah sinyal 'klik' agar tampak seperti instruksi amat diperlukan. Seperti terlihat pada Gambar 1, Gambar 2, dan Gambar 4, sebuah 'klik' bertahan kurang lebih selama 100 ms, dan sinyal akustik memiliki 2 puncak yang berbeda jelas – berhubungan dengan waktu penekanan tombol dan waktu pelepasan tombol. Ada waktu jeda sesaat antara waktu puncak penekanan dan waktu puncak pelepasan.

Distribusi frekuensi paling baik dilakukan pada waktu puncak. Dihitung distribusi frekuensi ketika waktu puncak penekanan karena frekuensi waktu puncak pelepasan dianggap lebih rendah. Setelah menghitung distribusi frekuensi ketika waktu puncak penekanan, hasil hitungan tersebut kemudian dinormalisasi untuk menghasilkan nilai-nilai untuk *spectrum fall* antara 0 dan 1 seperti diperlukan oleh *neural network.*

Pada awalnya, digunakan nilai FFT [19] yang diekstraksi dari jendela 8-10 ms dari waktu puncak penekanan sebagai fitur. Hasil percobaan selanjutnya menghasilkan sebuah perbaikan. Ketika dilakukan perbesaran, ternyata puncak penekanan bisa dilihat terdiri dari dua interval aktif yang berbeda jelas pada saat awal dan akhir dari interval 10 ms, dengan waktu jeda relatif di tengahnya. Interval aktif ini berhubungan dengan ketika jari menyentuh tombol (puncak penekanan) dan ketika tombol menyentuh dasar papan kunci (puncak ketik). Dasar papan kunci bergetar untuk kedua kasus tersebut. Jika FFT diekstraksi dari jendela 2-3 ms berhubungan dengan salah satu dari dua interval aktif, hasil proses pengenalan meningkat beberapa persen. Alasan di balik hal ini adalah karena adanya *noise* di tengah-tengah interval 10 ms dan pada pinggir puncak ketik dan puncak penekanan yang menurunkan kualitas fitur. Puncak penekanan dapat diekspresikan lebih baik daripada puncak ketik pada banyak kasus bunyi 'klik'. Oleh karena itu digunakan waktu puncak penekanan untuk mengekstraksi fitur.

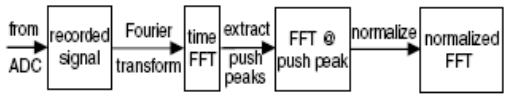
Tabel 1. Nilai ADCS untuk [0:9] kHz, gelombang radio, dan interval pergantian setiap 3 kHz.

kHz	0–9	3–3.4	0–3	1–4	2–5	3–6	4–7	5–8	6–9
ADCS	1.65	2.70	2.76	3.45	4.36	3.94	5.05	5.94	7.70

Detil tambahan mengenai ekstraksi fitur adalah tentang interval frekuensi yang masuk di dalam fitur itu sendiri. Dilakukan percobaan dengan fitur-fitur lain yang diekstraksi dari interval yang berbeda. Kemudian dialakukan perekaman terhadap *training set* dan *test set* selama 30 detik pada sebuah papan kunci PC tunggal. Untuk setiap interval frekuensi yang tersaring, dilakukan ekstraksi fitur, pelatihan *neural network*, menjalankan *neural network* pada *test set*, dan mengamati hasil proses pengenalan yang ada.

Tabel 1 menunjukkan ADCS untuk interval yang berbeda. ADCS adalah pengukuran yang memberikan posisi rata-rata dari simbol benar pada *ordered set* yang dikembalikan oleh *neural network*. Parameter ADCS dapat diinterpretasikan sebagai berikut: ADCS=1 artinya sebuah pengenalan tanpa *error* sama sekali, ADCS=15 (separuh dari jumlah tombol dalam percobaan) artinya tidak ada informasi yang diperoleh dan proses pengenalan sama sekali gagal. Ditemukan bahwa hasil proses pengenalan yang terbaik diperoleh dengan mengikut sertakan seluruh interval aktif pada saat ekstraksi fitur, kesimpulan ini diperoleh karena interval yang pendek menghasilkan proses pengenalan yang lebih buruk.

Pengamatan lain yang diperoleh selama melakukan percobaan adalah bahwa frekuensi yang lebih tinggi secara umum memiliki informasi yang lebih sedikit. Yang menarik adalah pada interval 300-3400 Hz gelombang suara telepon. Nilai ADCS yang relatif baik untuk interval ini menunjukkan bahwa 'pencurian dengar' pada bunyi 'klik' melalui telepon adalah suatu hal yang mungkin.



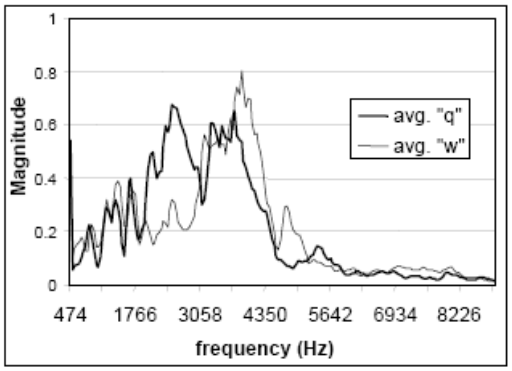
Gambar 5. Ekstraksi fitur.

Gambar 5 di atas menyimpulkan urutan transformasi yang diaplikasikan pada suara mentah dari bunyi 'klik' untuk kemudian dilakukan ekstraksi fitur.

Selain cara yang telah dijelaskan, dapat juga digunakan cara ekstraksi fitur alternatif lainnya. Misalnya, dalam suatu percobaan lain dapat saja menggunakan *cepstrum* daripada FFT [19] mentah. Faktanya, dapat saja dilakukan ekperimen dengan beragam pengklasifikasian, seperti sebuah *support vector machine* atau sebuah *decision tree* [14]. Seperti dapat dilihat, percobaan yang akan dilakukan cukup dapat mendemonstrasikan kerentanan dari papan kunci dan peralatan dengan tombol tekan lainnya berdasarkan serangan yang dilakukan terhadap bunyi 'klik' dari tombol tersebut.

2.3. Membedakan Dua Tombol

Sebelum mengaplikasikan *neural network* untuk membedakan dua tombol papan kunci berdasarkan bunyi 'klik' yang dihasilkan masing-masing tombol, pertama perlu divisualisasikan perbedaan yang ada antara fitur yang diekstraksi dari suara-suara 'klik'. Caranya adalah dengan mengaplikasikan agregasi terhadap fitur-fitur yang dihasilkan pada jendela 10 ms dari sebuah puncak penekanan untuk kedua tombol. Namun demikian, fitur-fitur yang diekstraksi dari jendela 2-3 ms dari sebuah puncak ketikan telah secara visual berbeda tanpa harus dilakukan agregasi (lihat Gambar 6). Perlu dicatat bahwa perbedaan visual antara puncak-puncak spektrum sentuhan berbeda untuk masing-masing tombol.



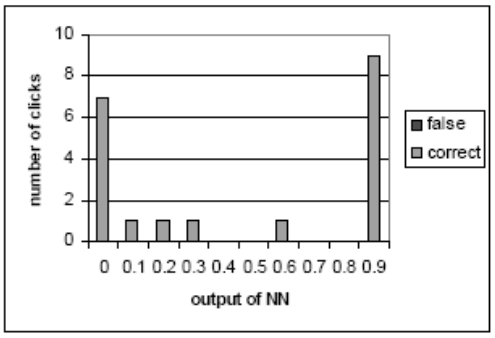
Gambar 6. Perbandingan terhadap spektrum rata-rata yang telah dinormalisasi (diekstraksi dari puncak-puncak ketik bunyi-bunyi 'klik').

Berikut adalah hasil dari proses pengenalan oleh *neural network*. Dipilih tombol *k* dan *l* dari papan kunci QWERTY standar untuk percobaan. Percobaan ini meliputi langkah-langkah sebagai berikut:

1. Menyiapkan pasangan {tombol, fitur} sebagai *training set* untuk *neural network*. Langkah ini meliputi 100 bunyi 'klik' dari tiap tombol yang kemudian diekstraksi fiturnya. Kecuali diberi catatan, bunyi 'klik' direkam pada jarak 0.5 meter.
2. Melatih *neural network* dengan pasangan {tombol, fitur}.
3. Menyiapkan fitur-fitur untuk melakukan tes terhadap *neural network* yang telah dilatih. Langkah ini meliputi

perekaman atas sebuah *test set* (10 bunyi ‘klik’ per tombol) dan mengekstraksi fitur-fiturnya.

4. Mengetes *neural network*. Pada langkah ini, ke *neural network* diberikan *test set* fitur dan hasil keluaran dari *neural network* itu dibandingkan dengan tombol yang sesungguhnya ditekan.



Gambar 7. Hasil dari proses pengenalan 10 bunyi ‘klik’ dari tombol *k* dan 10 bunyi ‘klik’ dari tombol *l*.

Gambar 7 di atas menunjukkan sebuah hasil percobaan dari pengaplikasian sebuah *neural network* terlatih untuk mengenali 10 bunyi ‘klik’ yang diproduksi oleh masing-masing tombol *k* dan *l*. *Neural network* mengenal bahwa sebuah bunyi ‘klik’ dihasilkan oleh tombol *k* (*l*) jika ia menghasilkan node yang bernilai antara 0 dan 0.5 (0.5 dan 1). Histogram menunjukkan jumlah dari pengenalan yang benar dan salah untuk tiap interval 0.1 dari jangkauan nilai keluaran *node*. Pada Gambar 7, tidak ditemukan pengenalan yang salah, yang artinya kesemua 20 bunyi ‘klik’ dikenali dengan benar.

Rata-rata hanya terdapat 0.5 pengenalan yang salah tiap 20 bunyi ‘klik’, yang menunjukkan bahwa serangan jenis ini dapat menjadi amat efektif dan berfungsi dengan baik.

2.4. Efek Jarak Perekaman

Dalam percobaan di atas, semua bunyi ‘klik’ direkam dari jarak pendek yang kurang dari 1 meter. Oleh karena itu perlu dilakukan percobaan untuk membedakan dua tombol dengan jarak rekam bunyi ‘klik’ yang beragam untuk mempelajari pengaruh jarak terhadap kualitas dari proses pengenalan. Untuk perekaman jarak jauh ini digunakan *parabolic microphone* yang murah. *Microphone* diletakkan di belakang orang yang sedang mengetik. Orang tersebut duduk di sebuah ruang berbentuk kubus, dengan suara *noise* dari latar belakang yang terbilang kecil.

2.5. Tombol Dalam Jumlah Banyak

Kemudian juga dilakukan pengamatan atas pengaruh tombol dalam jumlah banyak terhadap kualitas dari proses pengenalan.

Pelatihan terhadap *neural network* dilakukan untuk mengenali 30 tombol pada sebuah papan kunci (‘q-p’,’a-;’,’z-/’). Kemudian dilakukan perekaman sebanyak 10 bunyi ‘klik’ untuk tiap tombol. *Neural network* dilatih untuk mengenal sebuah tombol dengan memberikan nilai yang mendekati 1 untuk n*ode* keluaran unik, dan semua *node* lain diberi nilai mendekati 0. Pada saat dilakukan tes, tombol tertentu menjadi sulit dikenali jika *node* keluaran yang bersangkutan diberikan nilai terbesar ketika fitur yang dimiliki tombol bersangkutan diberikan sebagai *input* pada *neural network.*

Tabel 2. Neural network yang dilatih dengan 300 bunyi ‘klik’, 10 bunyi ‘klik’ tiap tombol.

Keyboard A, ADCS: 1.99						
key pressed	q	w	e	r	t	y
recognized	9,0,0	9,1,0	1,1,1	8,1,0	10,0,0	7,1,0
key pressed	u	i	o	p	a	s
recognized	7,0,2	8,1,0	4,4,1	9,1,0	6,0,0	9,0,0
key pressed	d	f	g	h	j	k
recognized	8,1,0	2,1,1	9,1,0	8,1,0	8,0,0	8,0,0
key pressed	l	;	z	x	c	v
recognized	9,1,0	10,0,0	9,1,0	10,0,0	10,0,0	9,0,1
key pressed	b	n	m	,	.	/
recognized	10,0,0	9,1,0	9,1,0	6,1,0	8,1,0	8,1,0

Hasil dari percobaan ini dapat dilihat pada Tabel 2. Untuk tiap 10 bunyi ‘klik’ yang direkam dari satu tombol dapat diberikan tiga angka: berapa kali *node* berkorespondensi terhadap tombol yang memiliki nilai terbesar, nilai kedua terbesar, dan nilai ketiga terbesar di antara 30 *node* yang ada. Dapat diamati bahwa jika *node* yang berkorespondensi terhadap tombol yang ditekan tidak memiliki nilai yang terbesar di antara 30 *node*, maka dengan kemungkinan terbesar, *node* tersebut memiliki nilai terbesar kedua atau terbesar ketiga dari keseluruhan *node* yang ada.

Sebagai kesimpulan, sebuah tombol dapat dikenali benar dengan nilai terbesar diberikan pada *node* yang benar sebanyak 79% dari keseluruhan 300 tes bunyi ‘klik’. Kemudian sebanyak 7% diperoleh hasil pengenalan yang benar dengan nilai terbesar kedua dan 2% pada nilai terbesar ketiga diberikan pada *node* bersangkutan yang benar. Jadi, tombol benar yang tidak ditemukan oleh *neural network* berdasarkan tiga kandidat yang diajukan hanya

sebanyak 12%. Percobaan ini kembali menekankan kuatnya serangan jenis ini.

2.6. Papan Kunci PC Dalam Jumlah Banyak

Selanjutnya diteliti kemungkinan terjadinya serangan dengan *neural network* yang dilatih dengan papan kunci yang berbeda namun memiliki jenis yang sama. Percobaan dilakukan dengan menggunakan tiga papan kunci GE. Setelah melatih *neural network* menggunakan papan kunci A, *neural network* yang sama diaplikasikan untuk melakukan proses pengenalan pada papan kunci B dan C.

Tabel 3. Penyerangan terhadap papan kunci B dan C menggunakan neural network yang dilatih dengan papan kunci A yang berjenis sama. Ada 300 bunyi 'klik' untuk tiap papan kunci, dengan 10 bunyi 'klik' untuk tiap tombol.

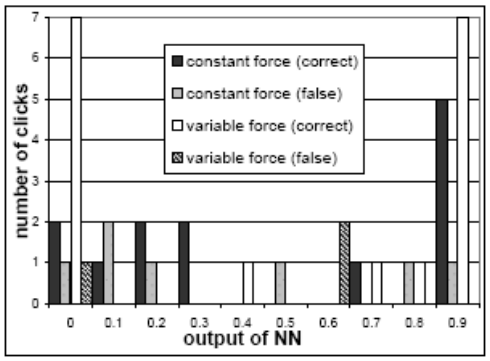
Keyboard B, ADCS: 9.24						
key pressed	q	w	e	r	t	y
recognized	6,1,1	4,1,1	0,1,0	0,2,1	5,1,1	1,0,0
key pressed	u	i	o	p	a	s
recognized	1,2,1	4,1,1	4,3,1	4,1,1	4,1,0	2,1,0
key pressed	d	f	g	h	j	k
recognized	1,4,0	0,0,0	1,0,1	5,1,1	9,0,0	1,0,2
key pressed	l	;	z	x	c	v
recognized	5,0,1	3,2,0	1,0,2	0,0,0	2,0,0	0,2,2
key pressed	b	n	m	,	.	/
recognized	3,3,1	3,1,1	5,1,1	0,2,1	2,1,0	7,2,1
Keyboard C, ADCS: 9.10						
key pressed	q	w	e	r	t	y
recognized	1,1,3	0,0,1	0,0,1	4,3,1	0,0,0	0,0,0
key pressed	u	i	o	p	a	s
recognized	2,3,0	1,3,0	3,3,3	1,1,1	0,1,0	1,2,0
key pressed	d	f	g	h	j	k
recognized	2,0,1	0,1,0	2,0,4	2,4,1	0,3,1	3,1,0
key pressed	l	;	z	x	c	v
recognized	1,0,0	1,1,0	2,2,0	0,1,1	10,0,0	1,0,2
key pressed	b	n	m	,	.	/
recognized	7,1,1	7,1,1	5,0,2	1,1,3	4,1,0	2,1,1

Hasil dari percobaan ini dapat dilihat pada Tabel 3. Seperti dapat diterka sebelumnya, kualitas dari proses pengenalan ini mengalami kemunduran dibandingkan kasus di mana *neural network* digunakan untuk menyerang papan kunci yang sama ketika digunakan untuk latihan (lihat Tabel 2). Dapat kita lihat bahwa 28%, 12%, 7%, dan 5% dari bunyi 'klik' dapat dikenal dengan baik sebagai kandidat pertama, kedua, ketiga, dan keempat untuk papan kunci B. Jadi, selama tes tombol benar ditemukan sebanyak 52% dari keempat terkaan yang diberikan oleh neural network. Sedangkan untuk papan kunci C, statistik yang sama diperoleh sebesar 50%.

2.7. Menangani Gaya Pengetikan yang Berbeda

Pada percobaan sebelumnya, semua bunyi 'klik' yang digunakan dalam *training set* dan *test set* diciptakan oleh orang yang sama, menggunakan jari dan kekuatan pengetikan yang sama.

Kemudian dipelajari pengaruh dari proses pengenalan jika seseorang mengetik dengan kekuatan yang beragam. Pada awalnya, *neural network* dilatih dengan bunyi 'klik' yang dihasilkan oleh kekuatan pengetikan yang konstan. Hasil dari proses pengenalan terhadap bunyi 'klik' yang dihasilkan oleh kekuatan pengetikan yang sama amat rendah (lihat Gambar 8).



Gambar 8. Bunyi 'klik' yang dihasilkan dengan kekuatan pengetikan yang beragam kemudian diklasifikasikan menggunakan dua *neural network*. Pada satu *neural network* digunakan kekuatan pengetikan yang konstan sebagai *training set*. Sedang untuk neural network satunya digunakan kekuatan pengetikan yang beragam.

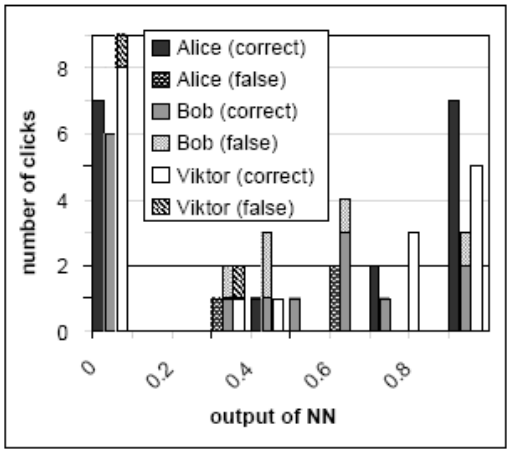
Selanjutnya, diciptakan sebuah *training set* di mana bunyi 'klik' dihasilkan oleh kekuatan pengetikan yang beragam dan melatih kembali *neural network*. Hasil dari proses tes ini (Gambar 8) sekarang sama baiknya dengan hasil dari percobaan awal, yang artinya neural network dapat dilatih untuk mengenali bunyi 'klik' yang dihasilkan oleh kekuatan pengetikan yang berbeda.

Percobaan lainnya menunjukkan bahwa kesimpulan yang sama juga berlaku untuk proses pengetikan dengan satu atau banyak jari. Secara singkat, jika *neural network* dilatih dengan gaya pengetikan satu jari, maka bunyi 'klik' yang dihasilkan oleh jari-jari yang lain dikenali dengan nilai *error* yang tinggi. Tapi *neural network* yang dilatih dengan banyak jari sama baiknya dengan *neural network* dasar:

kira-kira 1 bunyi ‘klik’ (dari keseluruhan 20 bunyi ‘klik’) dikenali dengan baik.

Juga dilakukan penelitian apakah gaya pengetikan yang berbeda dapat mempengaruhi kualitas dari proses pengetikan. Jawaban dari pertanyaan ini menjadi penting untuk serangan yang akan dilakukan, di mana *neural network* dapat dilatih oleh satu orang (si penyerang itu sendiri) untuk kemudian diaplikasikan pada papan kunci yang dipakai oleh orang lain.

Neural network yang dilatih menggunakan *training set* yang dihasilkan oleh kekuatan pengetikan yang beragam namun dilakukan oleh satu orang yang sama. Kamudian, *test set* direkam dari hasil pengetikan tiga orang yang berbeda. Ketiga orang itu dibebaskan untuk menggunakan gaya pengetikan masing-masing. Hasil dari proses pengetikan (Gambar 9) menunjukkan bahwa adalah mungkin bagi *neural network* untuk dilatih oleh satu orang kemudian mengaplikasikannya untuk menyerang papan kunci yang sama namun digunakan oleh orang yang berbeda. Perbedaan dari gaya pengetikan hanya sedikit mempengaruhi kualitas dari proses klasifikasi bunyi ‘klik’.



Gambar 9. Bunyi ‘klik’ yang dihasilkan oleh tiga orang berbeda kemudian diujicobakan pada *neural network* yang sebelumnya telah dilatih oleh orang keempat yang berbeda.

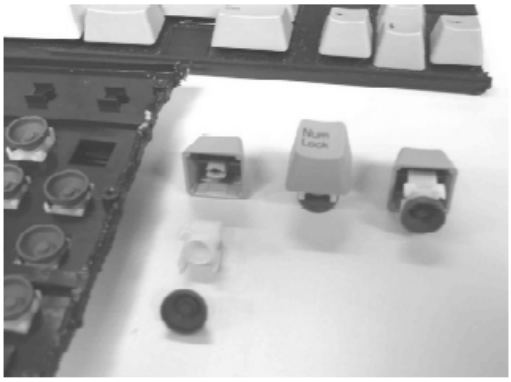
3. Kemungkinan Penangkalan Serangan

Kandidat yang paling jelas untuk menangkal serangan ini adalah dengan menggunakan sebuah *silent* papan kunci. Papan kunci jenis ini dapat terbuat dari karet atau sebuah papan kunci yang berbasiskan pada teknologi layar sentuh. Baru-baru ini, sebuah *virtual* papan kunci yang dapat diproyeksikan pada permukaan datar atau pada udara juga telah diproduksi.

Pilihan-pilihan ini memang lebih mahal daripada papan kunci mekanik yang standar. Mengetik pada papan kunci standar lebih nyaman daripada mengetik pada layar sentuh atau papan kunci yang terbuat dari karet.

Pada bagian ini, berdasarkan percobaan akan dicoba untuk menentukan mengapa bunyi ‘klik’ dari masing-masing tombol dapat berbeda. Kesimpulan yang diperoleh dari penelitian ini dapat membantu proses pendesainan papan kunci mekanik yang menghasilkan bunyi ‘klik’ yang relatif tak dapat dibedakan antara tombol satu dan tombol lainnya.

3.1. Papan Kunci Mekanik



Gambar 10. Arsitektur dari sebuah papan kunci mekanik.

Gambar 10 menunjukkan skema dari sebuah papan kunci mekanik. Tiap tombol terdiri dari tiga komponen: kepala tombol, komponen karet berbentuk kubah, dan sebuah komponen plastik yang menghubungkan kepala dan komponen karet tadi. Semua tombol ditahan oleh rangka plastik di papan kunci, dengan komponen plastik menembus rangka plastik ini.

Di bawah komponen karet terdapat sebuah *switch* elektrik yang berhubungan dengan tombol. Ketika tombol ditekan, komponen karet yang berbentuk kubah ikut tertekan, dan bagian atas dari kubah memaksa *switch* di bawahnya untuk menutup *circuit*.

3.2. Mengapa Bunyi ‘klik’ Berbeda Untuk Masing-masing Tombol

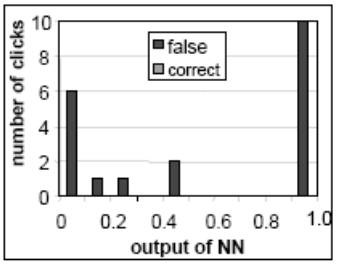
Diperoleh tiga hipotesis awal mengapa bunyi ‘klik’ berbeda untuk masing-masing tombol:

1. Interaksi dari suara yang dihasilkan oleh sebuah tombol dengan lingkungan

sekitar misalnya dengan tombol-tombol di sekitarnya.
2. Perbedaan (*microscopic*) amat kecil pada tombol.
3. Komponen yang berbeda pada rangka papan kunci mungkin menghasilkan suara yang berbeda ketika tombol yang berdekatan dengannya ditekan. Menganalogikan sebuah drum, menekan sebuah tombol pada lokasi yang berbeda pada rangka papan kunci menghasilkan pantulan suara yang berbeda.

Hipotesis pertama dibuktikan dengan dengan percobaan di bawah ini. Setelah *neural network* dilatih untuk membedakan dua tombol yang berbeda, lingkungan sekitar diubah dengan mencabut beberapa tombol tetangga dari papan kunci yang bersangkutan. Modifikasi ini seharunya mengubah bunyi 'klik' dari tombol tadi jika lingkungan sekitar menjadi pengaruh atas berbedanya bunyi. Namun demikian, *test set* yang direkam dengan baik diklasifikasikan oleh *neural network*, membantah hipotesis pertama.

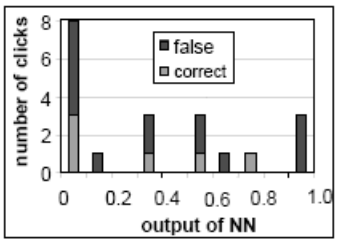
Untuk mengecek hipotesis yang kedua, *neural network* dilatih untuk mengenali dua tombol: *k* dan *l*. Kemudian rangka papan kunci diganti dengan yang lain dan dilakukan perekaman *test set*. Neural network mengidentifikasikan *k* sebagai *l* dan sebaliknya tombol *l* sebagai *k* (lihat Gambar 11).



Gambar 11. Hasil dari proses pengenalan terhadap dua tombol yang ditukar tempat pada rangka papan kunci yang sama. 10 bunyi 'klik' yang pertama dihasilkan oleh tombol *l*, sedangkan 10 bunyi 'klik' selanjutnya dihasilkan oleh tombol *k*. *Neural network* dilatih untuk mengenal tombol *k* sebagai 0 dan tombol *l* sebagai 1 sebelum kedua tombol ditukar tempatkan.

Percobaan ini menunjukkan bahwa hipotesis yang kedua dapat dibuktikan dan diperbaiki: Perbedaan (*microscopic*) amat kecil pada tombol tidak berperan atau memiliki peranan yang amat kecil dalam berbedanya suara yang dihasilkan oleh masing-masing tombol. Sebagai contoh, suara tombol spasi pada papan kunci mungkin berbunyi berbeda dengan tombol standar lainnya karena ukuran tombol yang berbeda.

Untuk mengecek hipotesis yang ketiga beberapa papan kunci dipasangkan masing-masing hanya satu tombol. Dengan jelas, operasi ini seharusnya menjelaskan hipotesis yang ketiga karena tidak ada catatan atas posisi-posisi tombol pada papan kunci. Jika hipotesis yang ketiga benar, maka seharusnya *neural network* tidak dapat membedakan bunyi 'klik' yang dihasilkan oleh masing-masing satu tombol pada satu papan kunci ini. Ternyata benar, setelah dilakukan proses pelatihan, neural network tidak dapat membedakan tombol berdasarkan bunyi 'klik' yang dihasilkannya (lihat Gambar 12), yang kemudian membuktikan kebenaran dari hipotesis ketiga.



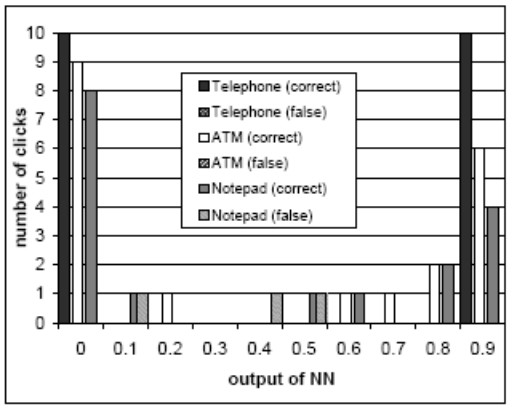
Gambar 12. Hasil dari proses pengenalan terhadap dua tombol yang dipotong dari rangka papan kunci. *Neural network* dilatih untuk mengenali tombol setelah tombol dipotong.

Percobaan ini menyatakan bahwa bunyi 'klik' pada masing-masing tombol berbeda karena masing-masing tombol diletakkan pada posisinya masing-masing pada sebuah papan kunci. Pengetikan pada posisi yang berbeda pada sebuah papan kunci menghasilkan bunyi yang berbeda. Baik perbedaan (*microscopic*) amat kecil atau lingkungan sekitar tidak mempengaruhi perbedaan bunyi yang dihasilkan oleh masing-masing tombol.

Konstruksi atas sebuah papan kunci yang memiliki bunyi yang sama antar tombolnya harus dirancang dengan memperhatikan hasil percobaan di atas. Kemungkinan yang bisa dilakukan, sebagai contoh, dengan tidak meletakkan semua tombol pada satu rangka papan kunci yang sama atau menciptakan papan kunci dari bahan yang tidak dapat menghasilkan getaran untuk menghindari rangka papan kunci berfungsi sebagai layaknya sebuah drum.

4. Papan Kunci Notebook, Set Tombol Nomor Telepon, dan Set Tombol Nomor ATM

Percobaan diulang untuk membedakan dua tombol dari *notebook* papan kunci, sebuah tombol nomor telepon, dan sebuah tombol nomor ATM. Untuk notebook papan kunci, 2 tombol yang dicobakan adalah *k* dan *l*. Tombol nomor telepon dan tombol nomor ATM dites dengan tombol nomor *1* dan *2*.



Gambar 13. Hasil dari proses pengenalan terhadap dua tombol pada *notebook* papan kunci (tombol *k* dan *l*) dan pada set tombol nomor telepon dan ATM (tombol nomor *1* dan *2*).

Hasilnya ditunjukkan pada Gambar 13 di atas. Dengan 2 pengenalan yang salah dari keseluruhan 20 bunyi 'klik', *notebook* papan kunci menunjukkan kerentanan terhadap serangan yang lebih rendah daripada papan kunci PC standar. Yang menarik, dari keseluruhan 20 bunyi 'klik' dari tombol nomor telepon dan tombol nomor ATM, semua dikenali dengan benar.

4.1. Set Tombol Nomor Telepon Dalam Jumlah Banyak

Percobaan diulang terhadap beberapa set tombol nomor telepon. Secara spesifik percobaan dilakukan dengan melatih neural network dengan satu set tombol nomor telepon untuk kemudian diaplikasikan terhadap set tombol nomor telepon yang lain.

Tabel 4. Set tombol nomor telepon A, B, dan C dari jenis yang sama diserang dengan neural network yang dilatih pada set tombol nomor telepon A. Ada 90 bunyi 'klik' yang dites untuk tiap set tombol nomor, dengan 10 bunyi 'klik' tiap tombol nomor.

Keypad A			Keypad B			Keypad C		
1	2	3	1	2	3	1	2	3
9,1,0	10,0,0	10,0,0	10,0,0	5,4,1	3,2,2	6,4,0	0,0,0	0,1,0
4	5	6	4	5	6	4	5	6
9,0,1	10,0,0	10,0,0	0,0,0	9,1,0	6,2,2	0,1,2	0,0,0	2,3,1
7	8	9	7	8	9	7	8	9
10,0,0	10,0,0	10,0,0	10,0,0	0,0,1	6,3,1	9,1,0	1,1,3	4,3,1
ADCS: 1.03			ADCS: 2.59			ADCS: 4.08		

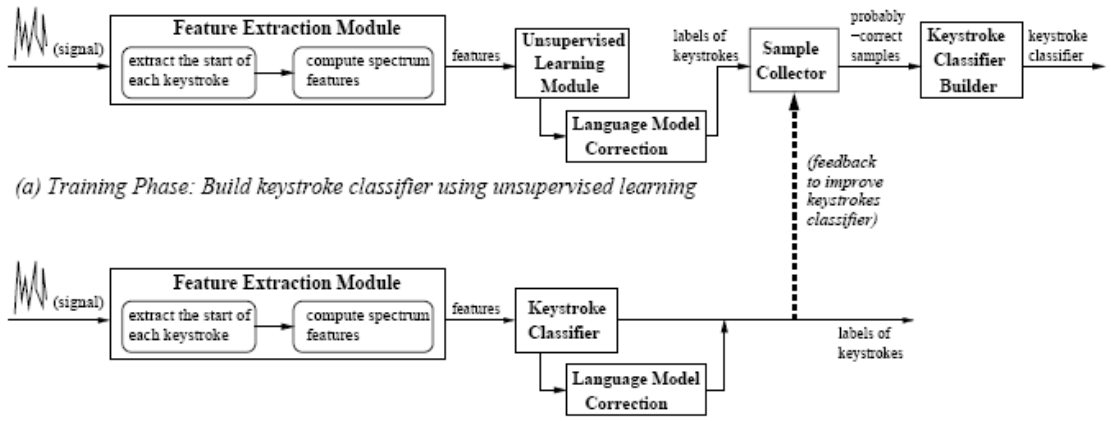
Data pada Tabel 4 di atas ditulis dengan format yang sama dengan Bagian 2.6. Kesimpulan dari percobaan ini mirip dengan hasil dari percobaan terhadap papan kunci:

1. Adalah mungkin untuk menyerang sebuah telepon menggunakan neural network yang dilatih dengan telepon lain yang memiliki tipe sama. Namun demikian, kualitas dari proses pengenalan ini lebih rendah dibandingkan kasus di mana neural network dilatih dengan telepon yang sama.
2. Kualitas dari proses pengenalan berbeda dari satu set tombol nomor ke set tombol nomor yang lain. Sebagai contoh, bunyi 'klik' dari telepon B dikenali dengan lebih baik daripada bunyi 'klik' dari telepon C.

5. Serangan Dengan Metode Lain

Pada tahun 2005, Li Zhuang, Feng Zhou, dan J.D. Tygar dari University of California, Berkeley melakukan peningkatan terhadap serangan yang sebelumnya dilakukan oleh Dmitri Asonov dan Rakesh Agrawal.

Pertama dilakukan perekaman terhadap seorang pengguna yang sedang mengetik teks berbahasa Inggris dengan sebuah papan kunci, kemudian diciptakan sebuah kakas pengenal yang dapat, dengan akurasi tinggi, menentukan tombol mana saja yang ditekan dari suara rekaman tadi jika diketik oleh orang yang sama, pada keyboard yang sama, di bawah kondisi ruang rekam yang sama. Kondisi ini dapat dengan mudah diperoleh misalnya dengan meletakkan sebuah *microphone* nirkabel pada daerah kerja pengguna atau dengan menggunakan *parabolic microphone*. Meskipun tidak dapat diketahui di awal apakah pengguna mengetikkan teks berbahasa Inggris, tapi dalam praktek dapat dilakukan perekaman terhadap pengetikan secara kontinu, pencobaan



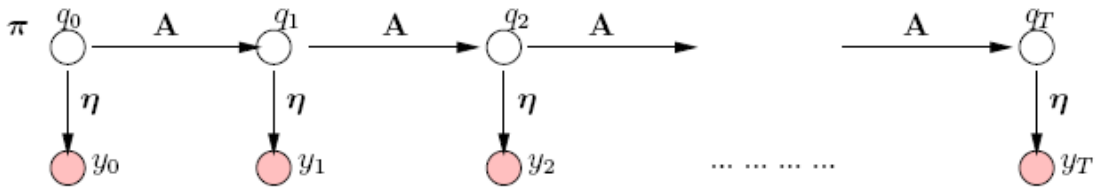
Gambar 14. *Overview* dari serangan.

serangan, dan kemudian dilihat apakah teks yang memiliki arti diperoleh.

Gambar 14 menunjukkan overview tingkat atas dari serangan.

Tahap (*phase*) pertama (Gambar 14(a)) melatih kakas pengenalan dengan cara:

1. Ekstraksi fitur. Digunakan fitur *cepstrum*, sebuah teknik yang dikembangkan oleh para peneliti di bidang pengenalan suara. Fitur *cepstrum* memberikan hasil yang lebih memuaskan daripada FFT.
2. Pengenalan tombol *unsupervised* menggunakan *training data set* yang tak berlabel. Dilakukan *clustering* untuk tiap tekanan ke dalam salah satu dari kelas K, menggunakan metode *data clustering* standar.
   Jika kelas *clustering* ini berkorespon dengan tepat terhadap semua tombol yang berbeda pada pemetaan satu-satu, maka dapat dengan mudah ditentukan pemetaan antara tombol dan kelasnya. Namun demikian, algoritma clustrering tidak pasti. Tekanan pada tombol yang sama terkadang diletakkan pada kelas yang berbeda dan sebaliknya tekanan tombol yang berbeda terkadang diletakkan pada kelas yang sama. Dinyatakan bahwa kelas adalah *random variable* atas dasar tombol sebenarnya yang ditekan. Tombol tertentu akan berada pada masing-masing kelas dengan kemungkinan tertentu. Dalam data yang ter-*clustering* dengan benar, kemungkinan dari satu atau banyak kelas akan mendominasi masing-masing tombol.
   Setelah kondisi distribusi telah ditentukan, dicoba untuk menemukan *sequence* tombol yang sesuai jika diberikan *sequence* kelas dari masing-masing tekanan. Umumnya, seseorang akan berpikir jika dicari huruf dengan kemungkinan terbesar untuk tiap tekanan maka akan menghasilkan perkiraan terbaik dan percobaan selesai. Tapi ada cara untuk melakukannya lebih baik, yaitu dengan menggunakan Hidden Markov Models (HMM) [7]. HMM memprediksi sebuah proses *stochastic* dengan kondisi (*state*). HMM menangkap hubungan antar tombol yang diketikkan dalam sebuah *sequence*. Sebagai contoh, jika tombol yang mungkin benar adalah *h* atau *j* (karena secara fisik letak kedua tombol itu bersebelahan dalam papan kunci) dan diketahui bahwa tombol yang sebelumnya ditekan adalah tombol *t*, maka tombol yang sedang ditekan kemungkinan terbesar adalah *h* karena *th* lebih umum dan lebih sering muncul daripada *tj*. Menggunakan hubungan seperti ini, baik tombol-tombol dan distribusi pemetaan tombol-ke-kelas secara efisien diperkirakan dengan algoritma HMM standar. Langkah ini menghasilkan akurasi yang lebih tinggi sebesar 60% untuk karakter dan 20% untuk kata.
3. Pengecekan ketik dan struktur kalimat. Digunakan koreksi ketik berdasarkan kamus dan sebuah model statistik sederhana dari struktur kalimat bahasa Inggris. Dua pendekatan ini, koreksi ketik dan struktur kalimat, dikombinasikan ke dalam sebuah Hidden Markov Model. Langkah ini menghasilkan akurasi yang lebih tinggi sebesar 70% untuk karakter dan 50% untuk kata. Pada tahap ini, teks dapat dibaca cukup jelas.



Gambar 15. Hidden Markov Model untuk pengenalan kunci *unsupervised*.

4. *Feedback-based training*. *Feedback-based training* menghasilkan sebuah pengklasifikasian tekanan yang tidak memerlukan sebuah model koreksi ketik dan struktur kalimat bahasa Inggris, mengizinkan pengenalan teks random, termasuk pengenalan kata kunci (*password*). Digunakan hasil koreksi atas hasil yang diperoleh sebelumnya sebagai *training samples* yang berlabel. Perlu dicatat bahwa koreksi yang dilakukan tidak 100% benar. Digunakan cara heuristik untuk memilih kata-kata yang kemungkinan besar adalah benar. Sebagai contoh, sebuah kata yang salah ketik atau kata yang berubah sedikit ketika langkah terakhir pengkoreksian lebih mungkin menjadi benar daripada yang melalui perubahan besar ketika pengkoreksian. Dalam percobaan, diambil kata-kata yang melalui pengkoreksian sebesar ¼ dari jumlah karakter dalam satu kata dan menggunakannya sebagai *training set* yang berlabel untuk melatih kakas pengklasifikasi. Tahap pengenalan (Gambar 14(b), akan dijelaskan kemudian di bawah) akan berusaha mengenali *training samples* lagi. Proses pengenalan yang kedua ini akan meningkatkan akurasi dari pengenalan tekanan. Digunakan jumlah pengkoreksian yang dilakukan pada tahap pengkoreksian ketik dan struktur kalimat sebagai indikator kualitas. Semakin sedikit koreksi dilakukan mengindikasikan semakin baiknya hasil yang diperoleh. Prosedur *feedback* yang sama dilakukan berulang hingga tidak ada peningkatan terlihat. Dalam percobaan, dilakukan tiga kali pengulangan proses *feedback*. Percobaan mengindikasikan bahwa campuran *linear classification* dan Gaussian dapat bekerja sebaik algoritma pengklasifikasian, dan lebih baik daripada neural network seperti yang digunakan dalam percobaan Dmitri Asonov dan Rakesh Agrawal. Dalam percobaan akurasi pengenalan karakter (tanpa tahap akhir dari koreksi ketik dan struktur kalimat) mencapai 92%.

Tahap kedua, tahap pengenalan, menggunakan kakas pengklasifikasi tekanan yang telah dilatih untuk mengenali suara rekaman baru. Jika teks mengandung kata random, seperti kata kunci, hasilnya adalah *output* langsung. Untuk teks bahasa Inggris, diaplikasikan pengkoreksian dan dilihat apakah teks yang masuk akal dapat dihasilkan.

Dalam praktek, seorang penyerang dapat menentukan apakah sebuah teks bersifat random. Seorang penyerang juga dapat mengidentifikasi kejadian-kejadian ketika seorang pengguna mengetik nama dan kata kunci. Sebagai contoh, pemasukkan kata kunci biasa mengikuti URL untuk website yang meminta masukkan kata kunci. Teks berarti yang diperoleh dari tahap pengenalan ketika penyerangan berlangsung dapat juga di-*fedback* ke dalam tahap pertama. *Sample* baru ini berikut dengan sample lama yang telah ada dapat digunakan bersama untuk mendapatkan akurasi yang lebih tinggi lagi. Rata-rata pengenalan meningkat seiring waktu.

Percobaan mengikutsertakan *data set* yang direkam di lingkungan yang berisik dan tenang serta menggunakan empat papan kunci yang berbeda (lihat Tabel 5 dan Tabel 6).
Contoh hasil dari percobaan ini adalah sebagai berikut. Teks yang dikenali dengan pengklasifikasi HMM, dengan fitur *cepstrum* (kata yang digaris bawah artinya salah):

> the big money fight has drawn the shoporo od dosens of companies in the entertainment industry as well as attorneys gnnerals on states, who fear the fild shading softwate will encourage illegal acyivitt, srem the grosth of small arrists and lead to lost cobs and dimished sales tas revenue.

Teks setelah melalui tahap pengkoreksian ketik:

Tabel 5. Rata-rata pemulihan teks pada tiap tahap. Semua nomor dalam persen.

		Set 1		*Set 2*		*Set 3*		*Set 4*	
		words	chars	words	chars	words	chars	words	chars
unsupervised learning	keystrokes	34.72	76.17	38.50	79.60	31.61	72.99	23.22	67.67
	language	74.57	87.19	71.30	87.05	56.57	80.37	51.23	75.07
1st supervised feedback	keystrokes	58.19	89.02	58.20	89.86	51.53	87.37	37.84	82.02
	language	89.73	95.94	88.10	95.64	78.75	92.55	73.22	88.60
2nd supervised feedback	keystrokes	65.28	91.81	62.80	91.07	61.75	90.76	45.36	85.98
	language	90.95	96.46	88.70	95.93	82.74	94.48	78.42	91.49
3rd supervised feedback	keystrokes	66.01	**92.04**	62.70	**91.20**	63.35	**91.21**	48.22	**86.58**
	language	**90.46**	**96.34**	**89.30**	**96.09**	**83.13**	**94.72**	**79.51**	**92.49**

Tabel 6. Rata-rata pemulihan teks pada tiap tahap. Dengan papan kunci yang beragam.

		Keyboard 1		*Keyboard 2*		*Keyboard 3*	
		words	chars	words	chars	words	chars
unsupervised learning	keystrokes	30.99	71.67	20.05	62.40	22.77	63.71
	language	61.50	80.04	47.66	73.09	49.21	72.63
1st supervised feedback	keystrokes	44.37	84.16	34.90	76.42	33.51	75.04
	language	73.00	89.57	66.41	85.22	63.61	81.24
2nd supervised feedback	keystrokes	56.34	88.66	54.69	86.94	42.15	81.59
	language	80.28	92.97	76.56	91.78	70.42	86.12
Final result	keystrokes	60.09	**89.85**	61.72	**90.24**	51.05	**86.16**
	language	**82.63**	**93.56**	**82.29**	**94.42**	**74.87**	**89.81**

```
the big money fight has drawn
the support of dozens of
companies in the entertainment
industry as well as attorneys
generals in states, who fear
the film sharing software
will encourage illegal
activity, stem the growth of
small artists and lead to lost
jobs and finished sales tax
revenue.
```

Teks asli. Perhatikan bahwa sebenarnya teks mengandung dua kesalahan ketik, salah satunya telah dikoreksi oleh proses pengkoreksian:

```
the big money fight has drawn
the support of dozens of
companies in the entertainment
industry as well as attorneys
gnnerals in states, who fear
the file sharing software
will encourage illegal
activity, stem the growth of
small artists and lead to lost
jobs and dimished sales tax
revenue.
```

6. Diskusi

6.1. Peningkatan kualitas serangan

Kedua serangan yang dibahas di atas belum mengikutsertakan tombol-tombol khusus seperti Shift, Control, Backspace, dan Capslock. Ada dua masalah di sini.

Satu adalah apakah tekanan dari tombol-tombol khusus itu dapat dipisahkan dari tekanan-tekanan tombol lain pada saat waktu pemrosesan sinyal. Percobaan awal membuktikan bahwa hal ini mungkin terjadi; puncak dari tekanan dengan mudah dipisahkan ketika melihat hasil rekaman.

Masalah yang satu lagi adalah bagaimana tombol modifikasi seperti Shift masuk ke dalam skema pengkoreksian ketik. Solusi ad hoc adalah dengan mengganti Shift atau Capslock dengan spasi. Backspace juga amat penting. Solusi yang ideal adalah dengan menerka teks akhir apa yang akan muncul setelah dilakukan *backspace*. Tapi mungkin hal tersebut akan memperumit algoritma pengkoreksian *error*. Jadi bisa saja seseorang mengenali tombol-tombol ini dan meninggalkan "kata" sebelum dan sesudah proses pengkoreksian *error* karena mungkin saja berupa kata yang tak lengkap.

Di sini sedikit bantuan manusia dapat berguna karena *backspace* relatif mudah dideteksi dengan telinga berdasarkan pada suara dan konteks, walau lebih sulit daripada spasi. Dengan mengasumsikan hal ini mungkin dilakukan, kakas pengklasifikasi dapat dilatih untuk mengenali tombol-tombol khusus ini dengan lebih akurat.

Pada percobaan selanjutnya, amat menarik untuk mencoba mendeteksi ketikan pada aplikasi tertentu, seperti pada sebuah *visual*

editor (misal emacs) atau lingkungan pengembangan perangkat lunak (misal Eclipse). Mengamati teks yang diketik pada lingkungan-lingkungan tersebut melahirkan tantangan baru karena mungkin lebih banyak tombol digunakan dan lebih banyak lagi tombol-tombol khusus yang digunakan. Teknik yang sama dipercaya dapat diaplikasikan untuk mengenali ketikan pengguna pada lingkungan-lingkungan ini serta meliputi beragam masukkan bahasa, termasuk bahasa alfabet kecil seperti bahasa Rusia atau Arab, bahasa alfabet besar seperti bahasa Cina atau Jepang, atau bahasa pemrograman.

Metode alternatif yang mungkin untuk prosedur *feedback training* adalah Hierarchical Hidden Markov Models (HHMMs). Pada HHMM, atau HMM yang berlipat ganda, tingkatan struktur kalimat dan koreksi ketik dibangun menjadi model tunggal. Algoritma untuk memaksimalkan kemungkinan *joint* global diasumsikan dapat memberikan hasil yang efektif seperti prosedur *feedback training*. Namun demikian pendekatan ini masih membutuhkan penelitian lanjut yang lebih mendalam.

Telah ditunjukkan bahwa nilai pengenalan lebih rendah pada lingkungan yang berisik. Serangan tidak akan berhasil sempurna jika, katakanlah, pengguna menjalankan musik ketika mengetik. Namun demikian, ada sebuah penelitian pada daerah pemrosesan sinyal yang dapat memisahkan suara dari kumpulan suara yang lain pada *channel* yang sama. Sebagai contoh, sistem karaoke yang rumit dapat memisahkan suara penyanyi dan suara musik latar belakangnya. Teknik yang sama dapat juga diaplikasikan di sini.

6.2. Pertahanan

Untuk membangun pertahanan terhadap serangan jenis ini, pertama harus dipastikan keamanan fisik atas mesin dan ruangan. Jika dibandingkan dengan teknologi modern *parabolic microphone* yang efektif, maka harus dipastikan bahwa tidak ada mesin pengganggu lain di dalam ruangan dan suara tidak dapat tembus ke luar ruangan. Penggunaan atas papan kunci yang bebas suara, seperti disarankan oleh Dmitri Asonov dan Rakesh Agrawal juga dapat mengurangi kerentanan. Namun demikian, kedua papan kunci 'diam' yang digunakan dalam percobaan terbukti tidak efektif dalam menangkal serangan.

Pesan yang lebih penting, bagaimana pun juga, adalah jangan hanya mengandalkan kata kunci atau bahkan *passphrase* yang panjang. Beberapa alternatif adalah dua faktor otentifikasi yang mengkombinasikan kata kunci dengan kartu cerdas, kata-kunci-satu-kali-pakai, otentifikasi biometrik, dan lain-lain. Namun demikian dua faktor otentifikasi tidak dapat menyelesaikan masalah secara tuntas. Teks yang diketik selain kata kunci juga menjadi berharga bagi penyerang.

Dmitri Asonov dan Rakesh Agrawal menyarankan agar pembuat papan kunci dapat membuat papan kunci yang memiliki bunyi sama untuk tiap tombolnya hingga tak dapat dibedakan satu dengan lainnya. Alasan dari berbedanya bunyi ini adalah dari rangka papan kunci yang berbunyi berbeda ketika ditekan pada daerah yang berbeda-beda. Jika hal ini benar, maka menggunakan rangka papan kunci yang lebih seragam mungkin dapat menangkal serangan. Namun demikian, tidak jelas apakah papan kunci jenis ini secara akan komersial tersedia. Masih ada kemungkinan kecil akan perbedaan sangat kecil yang mungkin masih ditangkap penyerang. Lebih lagi, papan kunci mungkin akan menghasilkan bunyi yang berbeda untuk tiap ketikan setelah satu bulan penggunaan.

7. Pekerjaan atau topik yang terkait

Sebuah artikel dari Computerworld [16] membahas keamanan komputer secara umum, dan menyatakan bahwa "*secrecy is an illusion*" dengan menyinggung beberapa cara eksotis untuk membobol sistem. "Trik papan kunci" dimasukkan sebagai salah satu pendekatan. Sayangnya, si penulis artikel tidak dapat memberikan referensi untuk "trik" ini [17].

Dokumen TEMPEST: NACSEM 5103, 5104, dan 5105 adalah dokumen tentang radiasi akustik, tapi (sayangnya) termasuk dokumen rahasia menurut dokumen NACSIM 5000 [15]. Dokumen ini juga menyatakan bahwa "papan kunci, mesin cetak, *relays* – alat ini menghasilkan suara, dan bisa menjadi sumber dari kerentanan", tanpa membahas lebih lanjut tentang masalah ini.

Pengarang dari [20] mengamati bahwa seorang pencuri dengar dapat mengkoleksi informasi waktu dari jalur sebuah sesi interaktif *shell* yang *secure*. Secara khusus, informasi waktu ini membongkar waktu jeda antar tombol yang diketik. Hal ini membuat sebagian informasi tentang indentitas dari tombol yang ditekan terbongkar kepada si pencuri dengar. Distribusi untuk pasangan tombol yang berbeda saling tumpang tindih, hingga informasi yang diperoleh relatif kecil. Lebih

lagi, pengguna yang berbeda mungkin menunjukkan waktu antar tekanan yang berbeda yang turut mengurangi nilai informasi yang dapat diperoleh. Namun demikian, dapat dibayangkan mengkombinasikan analisis waktu dengan serangan akustik yang dideskripsikan untuk membuat keputusan tentang bunyi 'klik' yang tidak secara ambigu dikenal berdasarkan pada data akustik sendiri. Perangkat lunak terkait – hanya serangan waktu – dijelaskan pada [21].

Papan kunci nirkabel dapat dicuri dengar dengan menggunakan *station* penerima yang lain. Untuk mencegah hal ini, beberapa pembuat papan kunci menawarkan enkripsi papan kunci dengan jenis *over-the-air*.

Dua serangan menggunakan rediasi elektromagnetik dari papan kunci secara sekilas dibahas pada [6]. Penulis artikel juga menjelaskan bagaimana menangkal serangan tersebut dengan memodifikasi *device driver* milik papan kunci, dan *firmware microcontroller* dari papan kunci. Radiasi dari LED, pada papan kunci LED khusus, dipelajari di [13]. Penggunaan suara yang dihasilkan oleh mesin *rotor* Hagelin untuk *side channel attack* telah didokumentasikan di [22]. Radiasi akustik untuk mesin cetak *matrix* ditunjukkan pada [7].

Penulis artikel dari [9] menangkal serangan ini dengan memberikan perlindungan pada pengguna dari pencuri yang menggunakan kamera video atau teropong untuk mencuri nomor telepon atau pin ATM dari jauh. Solusi yang ditawarkan adalah dengan menginstalasi *eye tracking system* pada terminal, jadi pengguna dapat menggunakan gerakan mata untuk memasukkan nomor.

8. Kesimpulan

Eksplorasi dilakukan terhadap radiasi akustik dari alat masukkan berbentuk papan kunci. Setelah memberikan deskripsi yang mendetil atas serangan dasar terhadap papan kunci PC, serangan kemudian diaplikasikan untuk alat masukkan dengan tombol tekan lainnya, seperti *notebook keyboard*, tombol nomor telepon, dan tombol nomor ATM.

Sebuah papan kunci bebas suara (non mekanik) adalah penangkal yang paling jelas dari serangan ini. Namun demikian, papan kunci jenis ini tidak lebih nyaman bagi pengguna juga tidak murah. Telah berhasil diidentifikasikan penyebab mengapa tombol memiliki bunyi yang berbeda satu dengan yang lainnya, hal ini berguna untuk mendesain papan kunci mekanik yang memiliki bunyi yang seragam untuk semua tombol.

Penelitian lebih lanjut dapat dilakukan atas permasalahan yang telah dibahas. Misalnya dapat memperbanyak variabel lingkungan untuk meneliti serangan mana yang berhasil pada lingkungan mana. Contoh lain adalah menyelidiki kerentanan dari jenis tombol nomor lainnya seperti tombol nomor kunci yang banyak ditemui pada banyak pintu rumah atau tombol nomor garasi. Hanya dengan mengukur dan menganalisa kerentanan dari alat penghasil suara inilah dapat dikembangkan penangkalnya demi meningkatkan faktor keamanan dari alat tersebut.

Jadi bidang kriptografi tidak hanya terkait dengan pembangunan algoritma yang rumit saja. Faktor keamanan juga penting untuk diperhatikan. Perlu dipikirkan cara agar kunci yang diperlukan dalam membuka *chipertext* didesain seaman mungkin, agar faktor keamanan dari sebuah algoritma kriptografi tak hanya kunci semata.

9. Daftar Referensi

[1] Canesta keyboards. http://www.canesta.com/products.htm.

[2] Chicklet keyboard from IBM PC Junior. http://www.digibarn.com/collections/devices/pcjr-chicklet-keyboard/.

[3] HoloTouch technology. http://www.holotouch.com.

[4] TouchStream keyboards. http://www.fingerworks.com/.

[5] The virtually indestructible keyboard. http://www.grandtec.com/vik.htm.

[6] R. J. Anderson and M. G. Kuhn. Soft tempest – an opportunity for NATO. In Proceedings of Protecting NATO Information Systems in the 21st Century, IST Symposium, Washington DC, USA, Oct. 1999.

[7] R. Briol. Emanation: How to keep your data confidential. In Symposium on Electromagnetic Security For Information Protection, SEPI'91, Rome, Italy, Nov. 1991.

[8] M. A. Casey. Introduction to MPEG-7: Multimedia Content Description Language, chapter Sound Classification and Similarity Tools. J. Wiley, 2001.

[9] M. D. Flickner, Q. Lu, and C. H. Morimoto. Gaze-based secure keypad entry system. Patent US6282553, 2001.

[10] M. G. Kuhn. Optical time-domain eavesdropping risks of CRT displays. In Proceedings of IEEE Symposium on

Security and Privacy, Berkley, California, USA, May 2002.
[11] M. G. Kuhn. Compromising emanations: eavesdropping risks of computer displays. Technical Report UCAM-CLTR-577, Computer Laboratory, University of Cambridge, 2003.
[12] M. G. Kuhn. Personal communication. Feb. 2004.
[13] J. Loughry and D. A. Umphress. Information leakage from optical emanations. ACM Transactions on Information and System Security, 5(3):262–289, Aug. 2002.
[14] T. Mitchell. Machine Learning. McGraw Hill, 1997.
[15] NSA. NACSIM 5000 TEMPEST fundamentals. National Security Agency, Fort George G. Meade, Maryland, http://cryptome.org/nacsim-5000.zip.
[16] N. Petreley. Secrecy is an illusion. Computerworld, http://www.computerworld.co.nz/webhome.nsf/0/7237CE66D15E3BFFCC256B8A00%0C087B, Apr. 2002.
[17] N. Petreley. Personal communication. Sept. 2003.
[18] R. Price, J. Willmore, and W. Roberts. Genetically optimised feedforward neural networks for speaker identification. Technical Report DSTO-TN-0203, Defence Science and Technology Organisation (Australia), 1999.
[19] S. W. Smith. The Scientist and Engineer's Guide to Digital Sound Processing. California Technical Publishing, 1997.
[20] D. X. Song, D. Wagner, and X. Tian. Timing analysis of keystrokes and SSH timing attacks. In Proceedings of 10th USENIX Security Symposium, Washington DC, USA, Aug. 2001.
[21] J. Trostle. Timing attacks against trusted path. In Proceedings of IEEE Symposium on Security and Privacy, Berkley, California, USA, May 1998.
[22] P. Wright. Spycatcher. Random House Value Pub, 1989.
[23] A. Zell, N. Mache, T. Sommer, and T. Korb. Recent developments of the SNNS neural network simulator. In Applications of Neural Networks Conf., SPIE, volume 1469, Orlando Florida, 1991.