Presented by:

Bruce Megget

Keyboard Acoustic Emanations

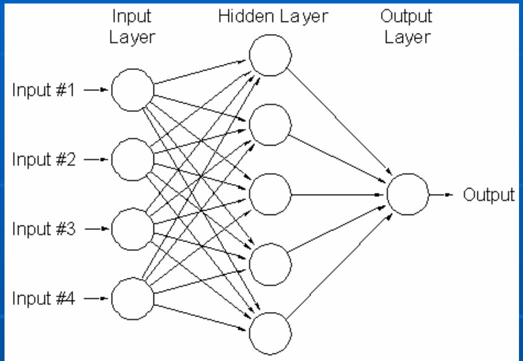
Dmitri Asonov and Rakesh Agrawal
“Keyboard Acoustic Emanations”
Symposium on Security and Privacy.
IEEE, February 2004.

Their premise

- Keyboard keys make unique sounds when pressed. ‘Eavesdropping’ on someone typing (by microphone) lets an attacker calculate what is being typed.

Background

- Neural Networks
  - Learns like a child does
  - The more it is taught, the more it will get assumptions correct
  - Can think of it as a brain



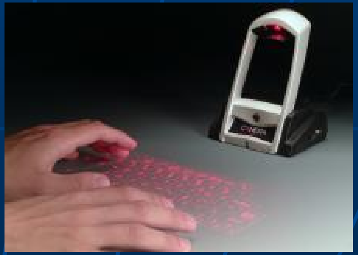
- What causes a unique click sound?
  - They believe each keyboard has a plate underneath the keys that acts like a drum
  - When a key is pressed, it makes a slightly different sound than another key at a different point on the plate
  - This is by no means the definitive reason key clicks are unique.
  - Is there a way to stop creating sounds when typing?

Possible Countermeasures

- Rubber keyboards
  - The Virtually Indestructible Keyboard (VIK) is made of a silicon based substance
- Touch screen keyboards
  - Screen recognizes pressure and calculates which button is being pressed by the position on the touch screen.
- Virtual keyboards
  - Projects a full sized keyboard onto a flat surface and tracks the movement of the users fingers to calculate which keys are being pressed

US$30

US$340



US$150

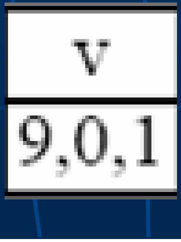
Issues I have found

- Appreciative
  - They provide a broad range of tests to investigate the effectiveness and reality of an attack.
    - These tests include:
      - Training and testing on one keyboard
      - Training on one keyboard and testing on another
      - Testing from different distances
      - Different typing styles
      - Testing on different key pads

- Critical
  - The amount of detail provided in their experiments is limited. They declare their intentions for the experiment and provide results, but do not list all the details. This makes it very hard to reproduce the results they attained.
  - They state that the accuracy of recognition for their system is higher than shown by their results. Some testing results do not give enough information to justify their high accuracy.

How they tested their system

- System tested with 30 keys from a QWERTY keyboard. These 30 keys correspond to 30 output nodes in the Neural Network
- Each of the 30 keys is clicked 10 times and the features input into the Neural Network for testing.
- The node with the highest value from the Neural Network is the key the Neural Network thinks has been pressed.
- Results from their tests are displayed in a table showing three numbers. How many times the correct node had the highest number, the second highest number and the third highest



v
9,0,1

Results from Testing

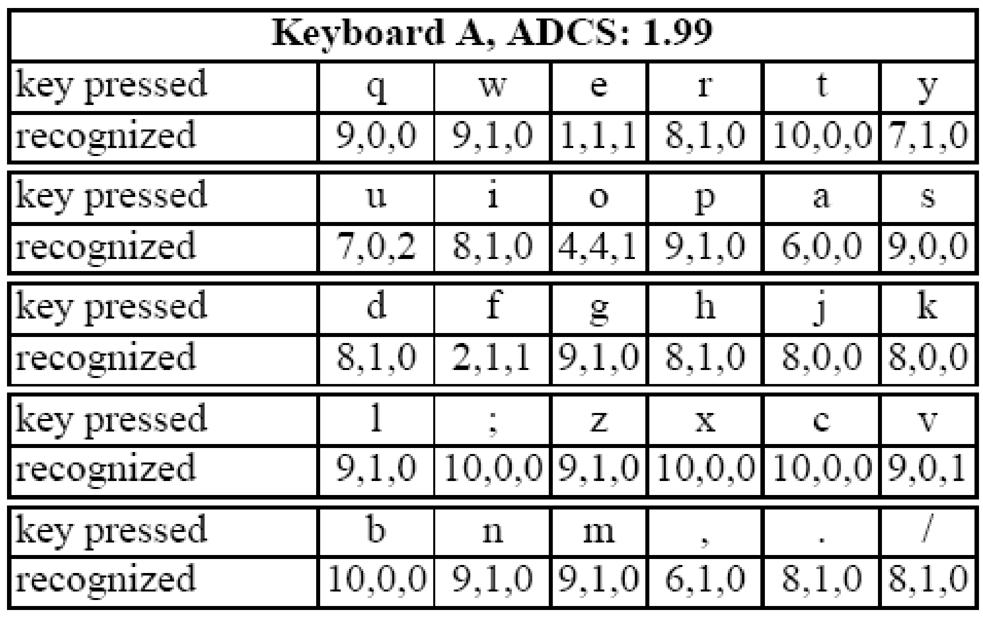
- Highest node formula:
  - (q1+ w1 + e1 + …)/300

    = 79% accurate
- Top 3 node formula:
  - (q1 + q2 + q3 + w1 + w2 + w3 + …)/300

    = 88% accurate

Where:

- q1 is the first number in the results for q,
- 300 is the number of clicks input into the system



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

- They say their results show:
  - That the system is 79% accurate at outputting the correct node (with the highest value) for the Neural Network.
  - That the system is 88% accurate when outputting one of the top three nodes from the Neural Network.
- How does their system pick the correct node out of the top three to justify this 88% accuracy?

Training on Keyboard A, and testing on Keyboards B and C

- Highest node formula:
  - (q1+ w1 + e1 + …)/300

    = 28% accurate for Keyboard B

    = 21% accurate for Keyboard C

- Top 3 node formula:
  - (q1 + q2 + q3 + w1 + w2 + w3 + …)/300

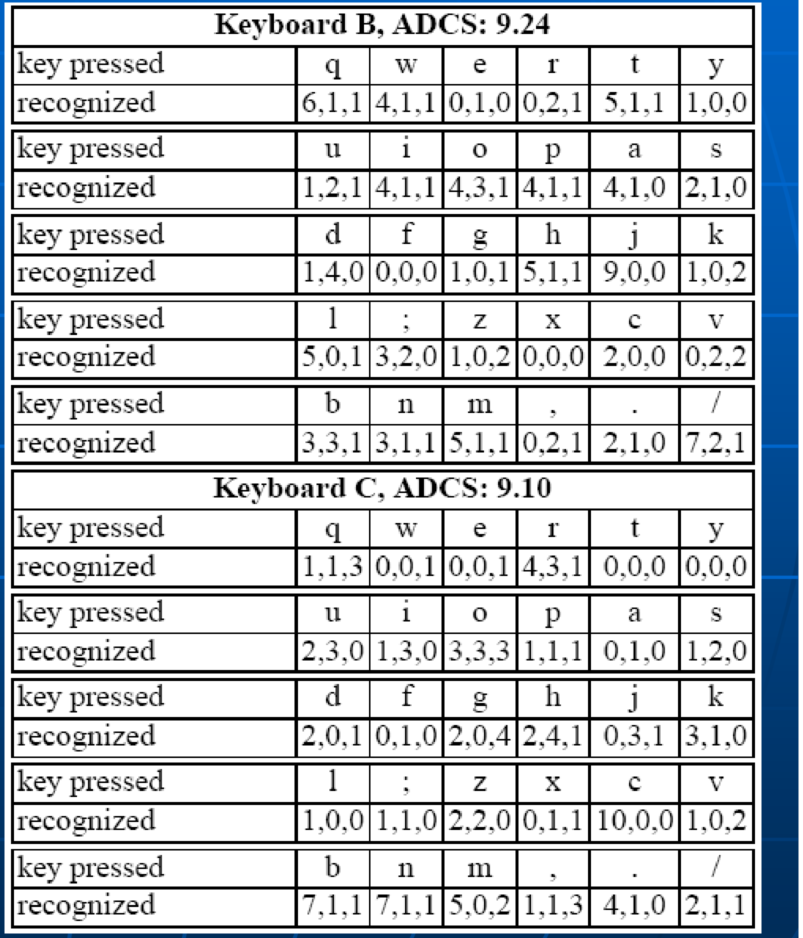
    = 46% accurate for Keyboard B

    = 42% accurate for Keyboard C

Where:

- q1 is the first number in the results for q,
- 300 is the number of clicks input into the system

- They state that (when looking at the top 4 recognized nodes) the system is:
  - 52% accurate on Keyboard B
  - 50% accurate on Keyboard C
- Why measure the accuracy on the top four nodes, and not the top three like before?
- We do not have the results for the top fourth output node so have to assume that the accuracy they state is correct.



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

For you, does the threat of being attacked/hacked justify the price of one of the possible countermeasures?