랜덤하게 입력되는 진동의 세기를 활용한 잠금 패턴 보안 강화구현

IT융합공학부
18211501
안규황

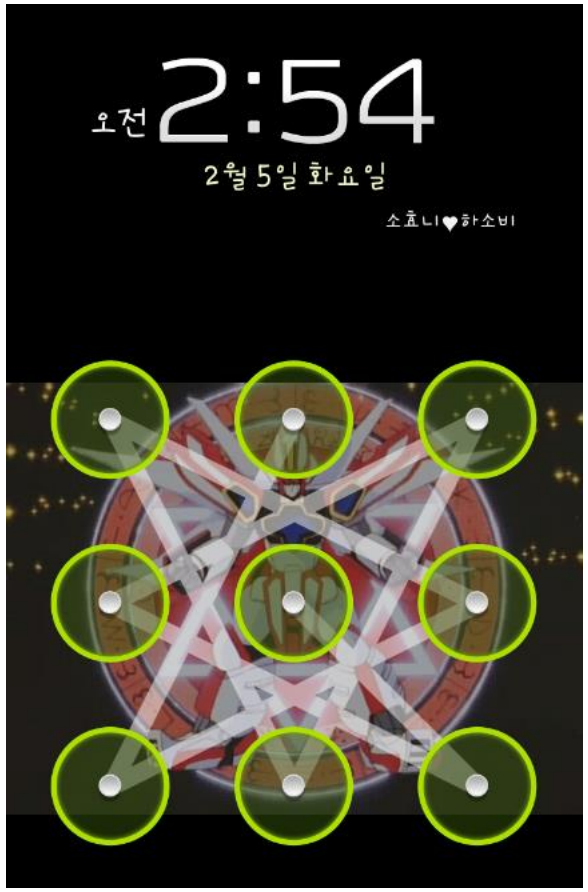
한성대학교 HANSUNG UNIVERSITY

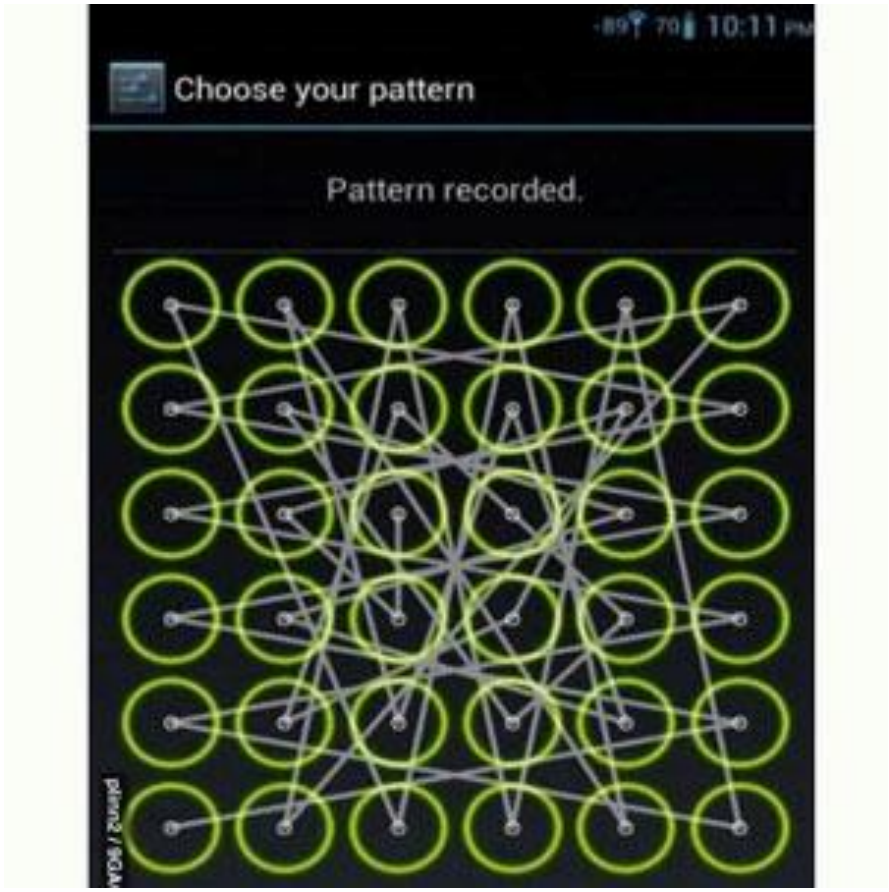
CryptoCraft LAB

Contents

CryptoCraft LAB

1 서 론





CryptoCraft LAB

1 서 론

패턴 잠금이란?

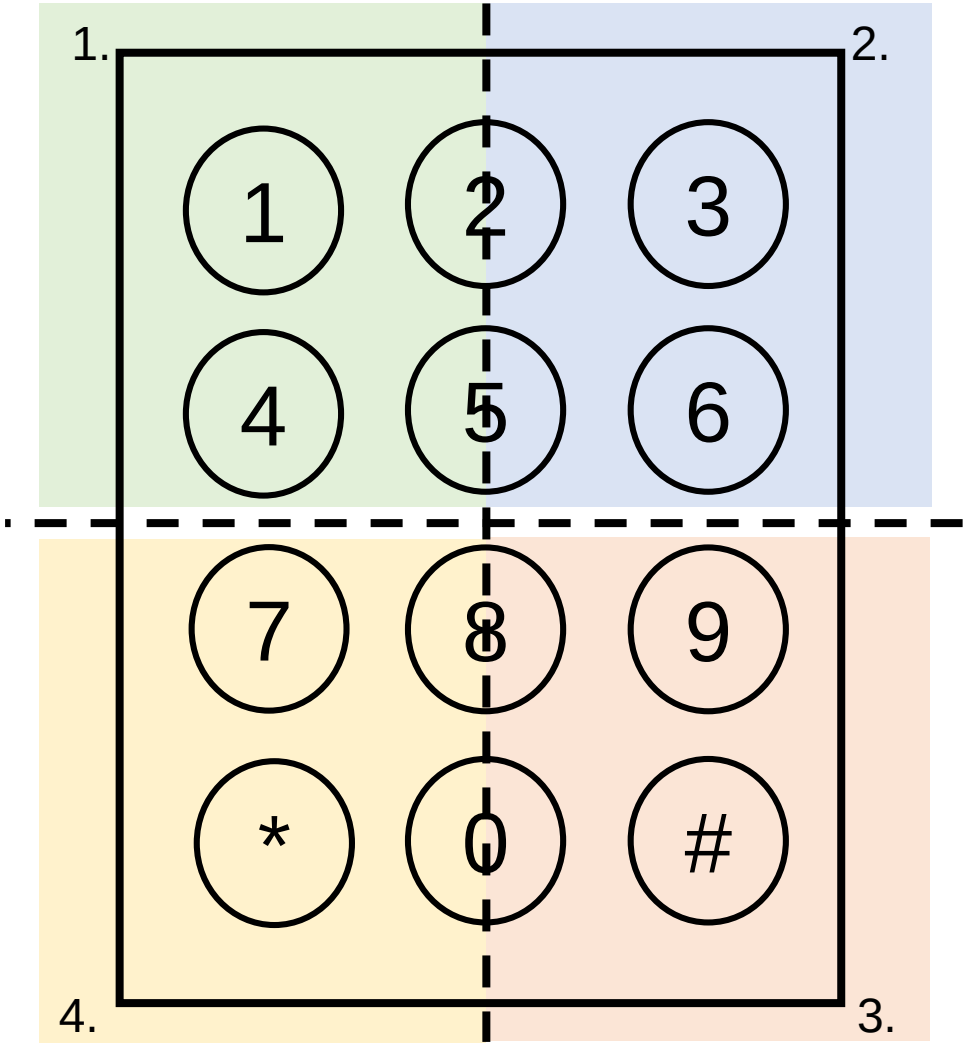
핸드폰의 보안을 위하여 암호를 설정하는 방법의 하나로,
핸드폰 화면의 **3x3점을 표시하고 점을 연결하는 패턴 모양을 미리 설정한 뒤 설정한 패턴으로 암호를 푸는 방식**

9개의 점을 사용자가 정한 순서대로 이어서 푸는 스마트폰 잠금 해제 방식이며,
안드로이드의 역사와 함께 시작

총9개점이 쓰이며 단순해 보여도 3x3 패턴으로 만들 수 있는
패턴의 경우의 수는 389,112가지

CryptoCraft LAB

2 패턴 공격 기법 – Shoulder Surfing Attack

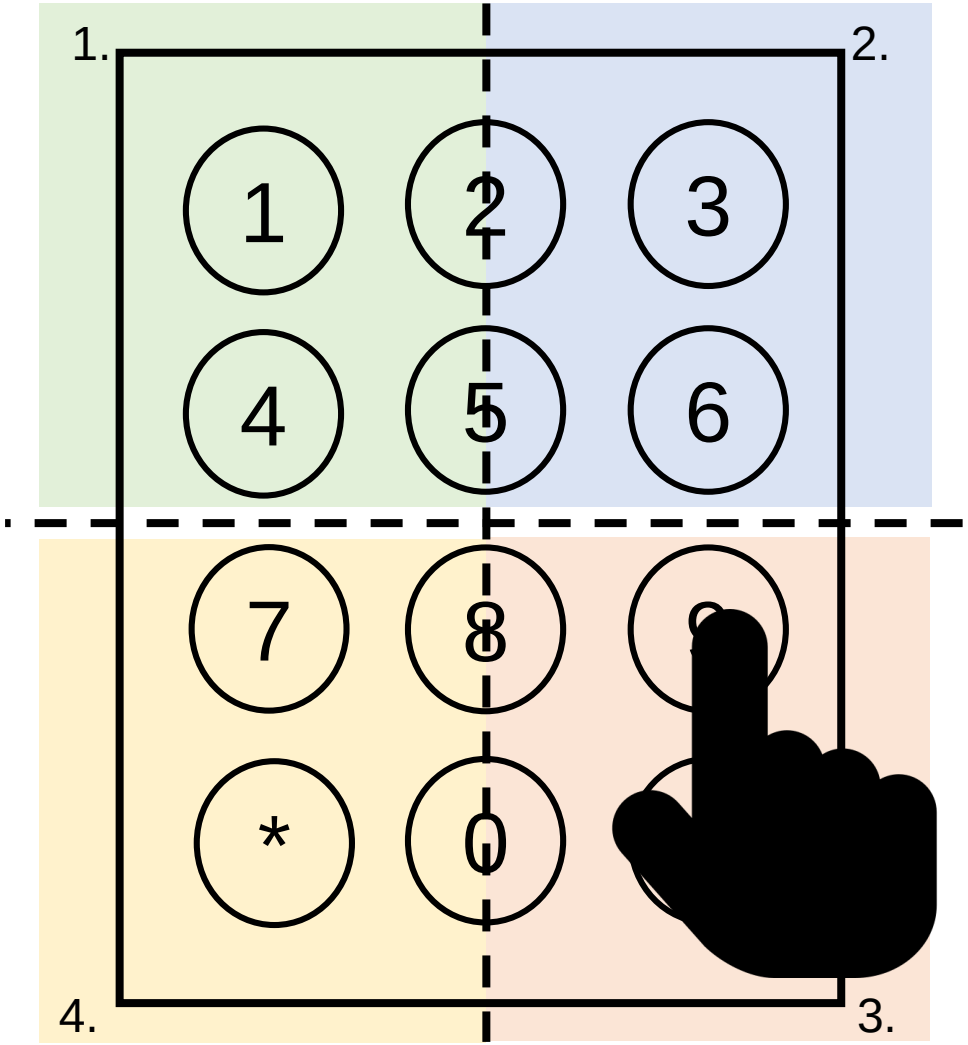


비밀번호를 누르는 손과 어깨의 움직임을 보고 유추하는 공격

‘1’을 누르는 손의 위치와 ‘9’를 누르는 손의 위치는 확연히 다름

CryptoCraft LAB

2 패턴 공격 기법 – Shoulder Surfing Attack

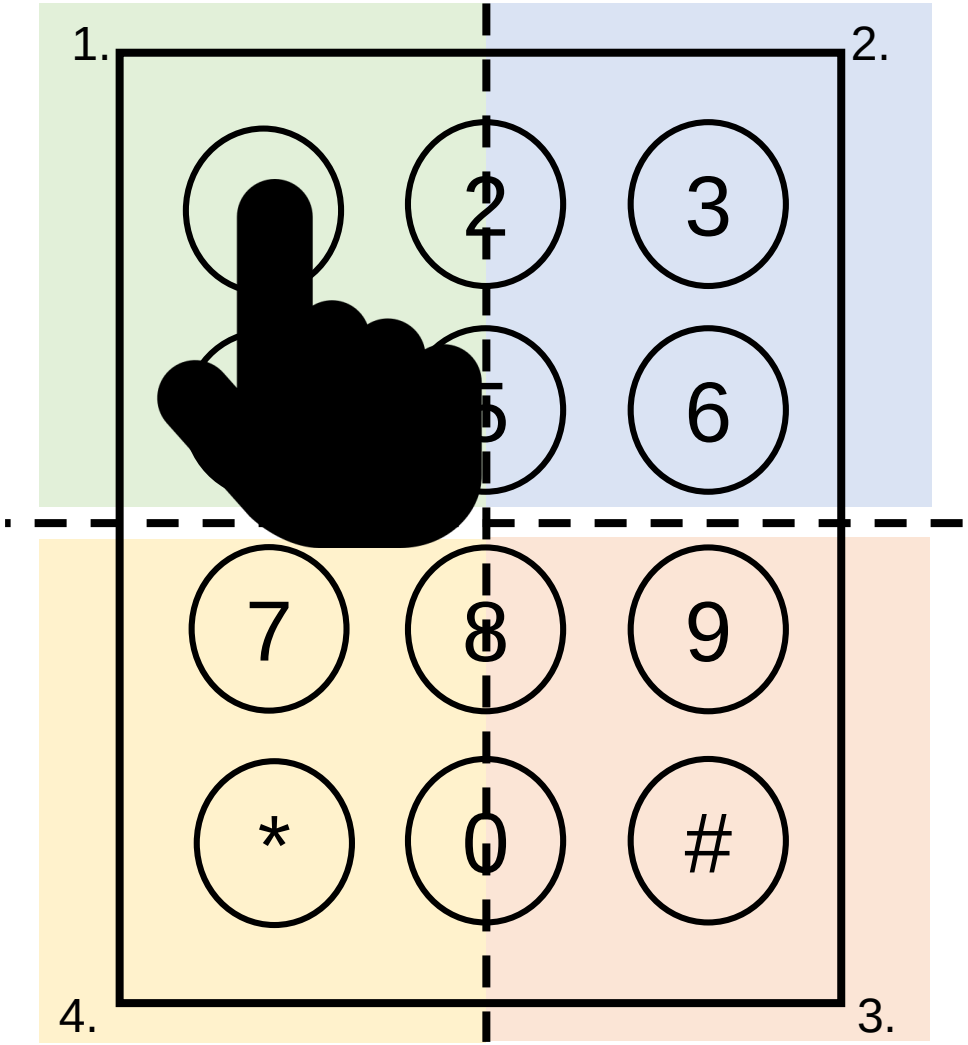


비밀번호를 누르는 손과 어깨의 움직임을 보고 유추하는 공격

‘1’을 누르는 손의 위치와 ‘9’를 누르는 손의 위치는 확연히 다름

CryptoCraft LAB

2 패턴 공격 기법 – Shoulder Surfing Attack

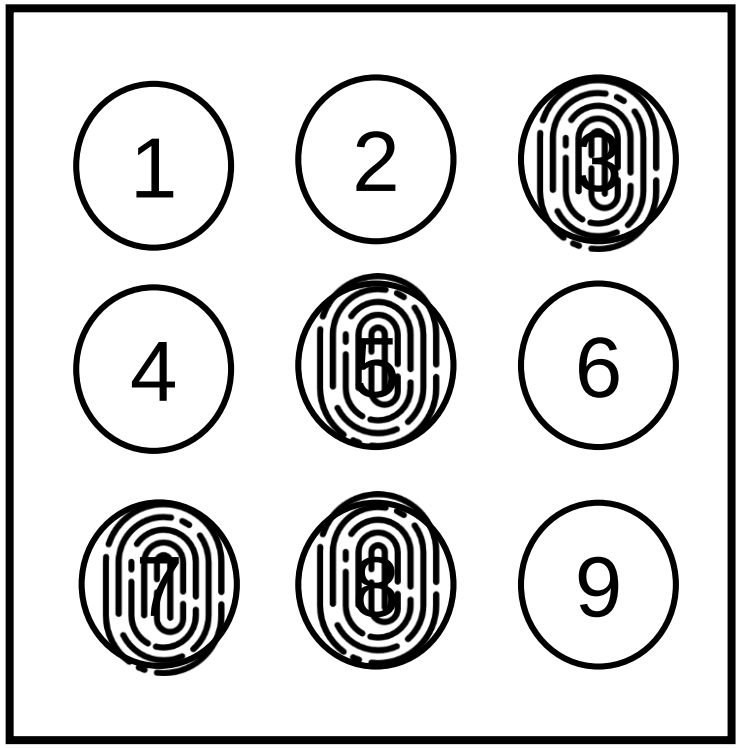


비밀번호를 누르는 손과 어깨의 움직임을 보고 유추하는 공격

‘1’을 누르는 손의 위치와 ‘9’를 누르는 손의 위치는 확연히 다름

CryptoCraft LAB

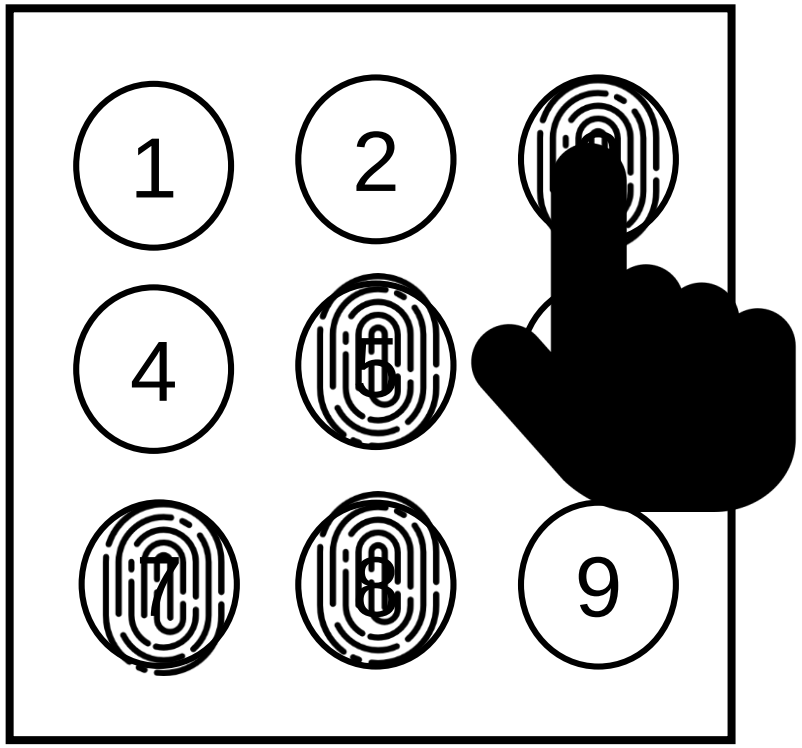
2 패턴 공격 기법 – Smudge Attack



3, 5, 7, 8에 지문이 묻어있는 것을 확인
4! 확률로 비밀번호 유추 가능

CryptoCraft LAB

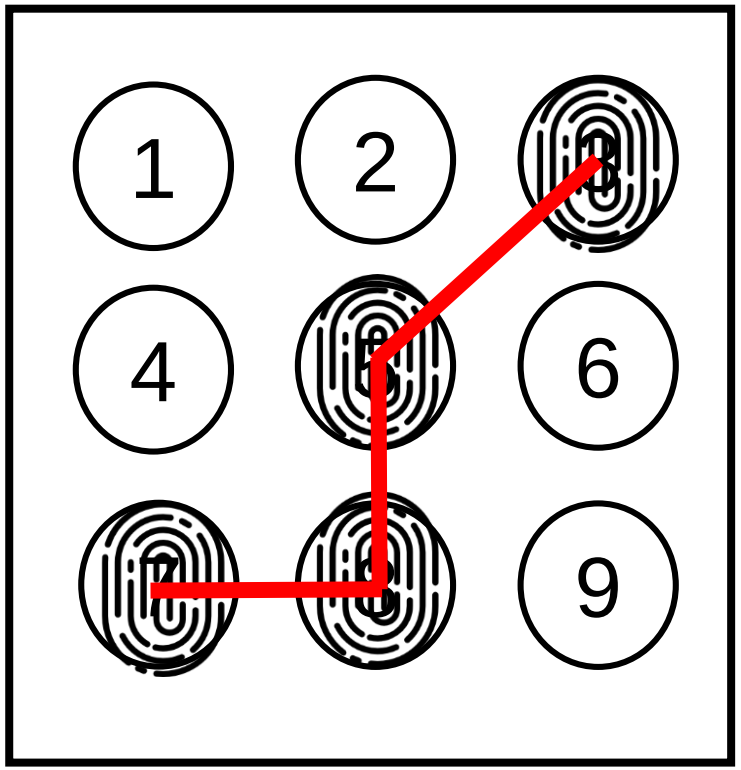
2 패턴 공격 기법 – Smudge Attack



3, 5, 8, 7 순으로 드래깅하여 잠금 해제를 했을 때,
남아있는 흔적을 이용하면 단 2가지 경우의 수로 잠금 해제 가능

CryptoCraft LAB

2 패턴 공격 기법 – Smudge Attack



3, 5, 8, 7 순으로 드래깅하여 잠금 해제를 했을 때,
남아있는 흔적을 이용하면 단 2가지 경우의 수로 잠금 해제 가능

CryptoCraft LAB

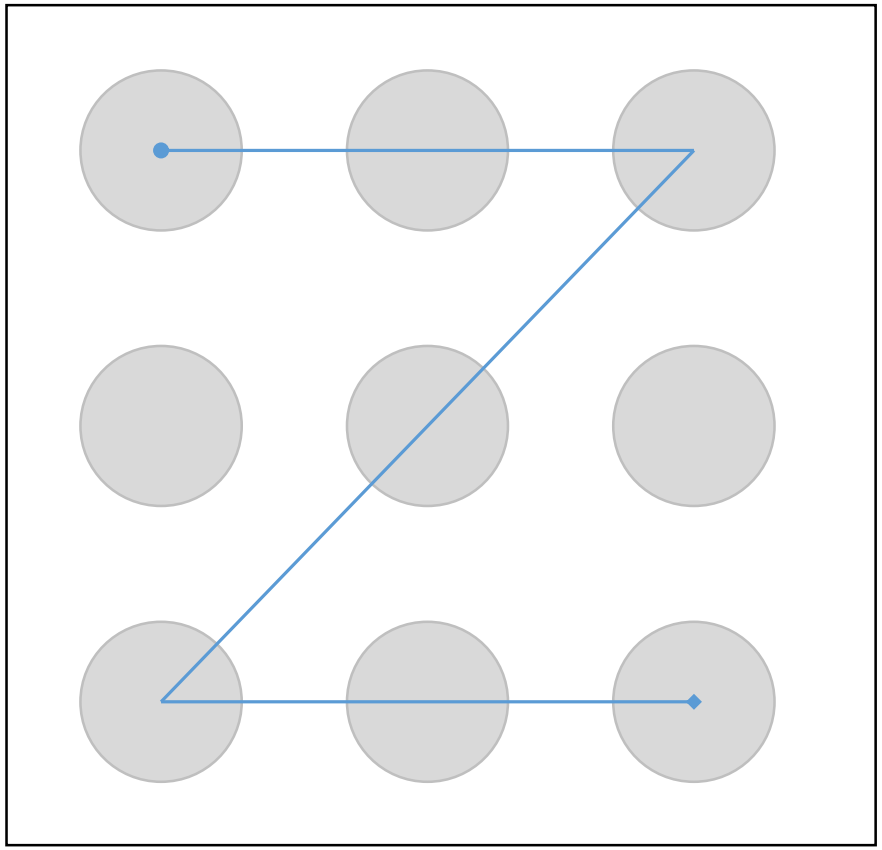
2 실제 사례

CryptoCraft LAB

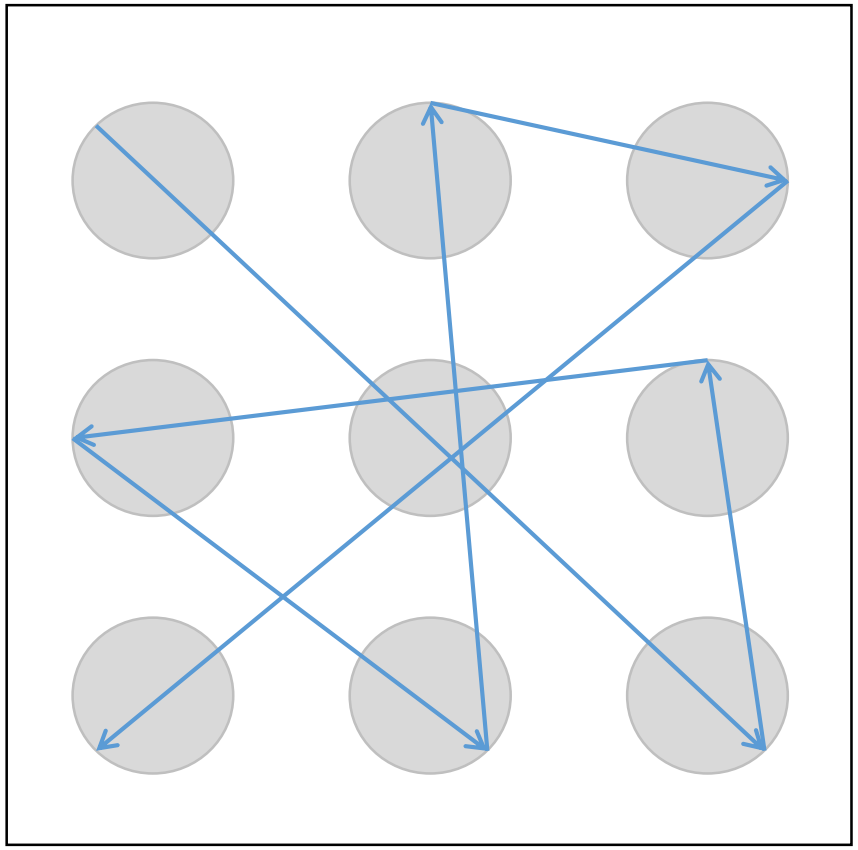
2 실제 사례

CryptoCraft LAB

2 패턴 잠금 - 복잡도



VS

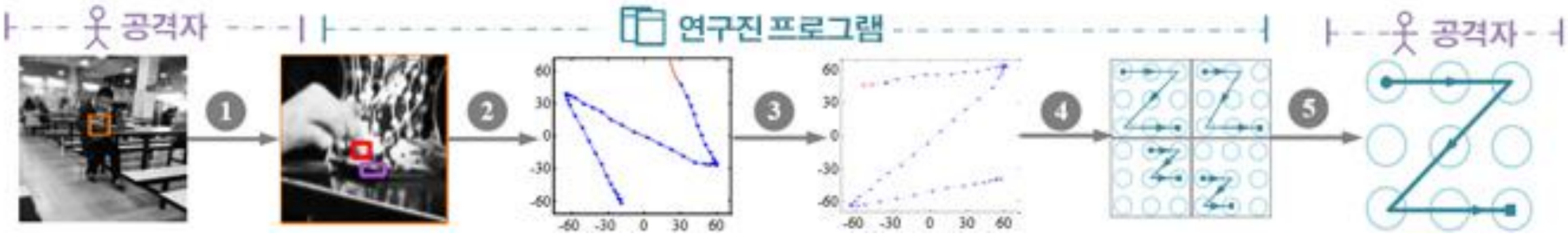


Simple

Complex

CryptoCraft LAB

2 패턴 잠금

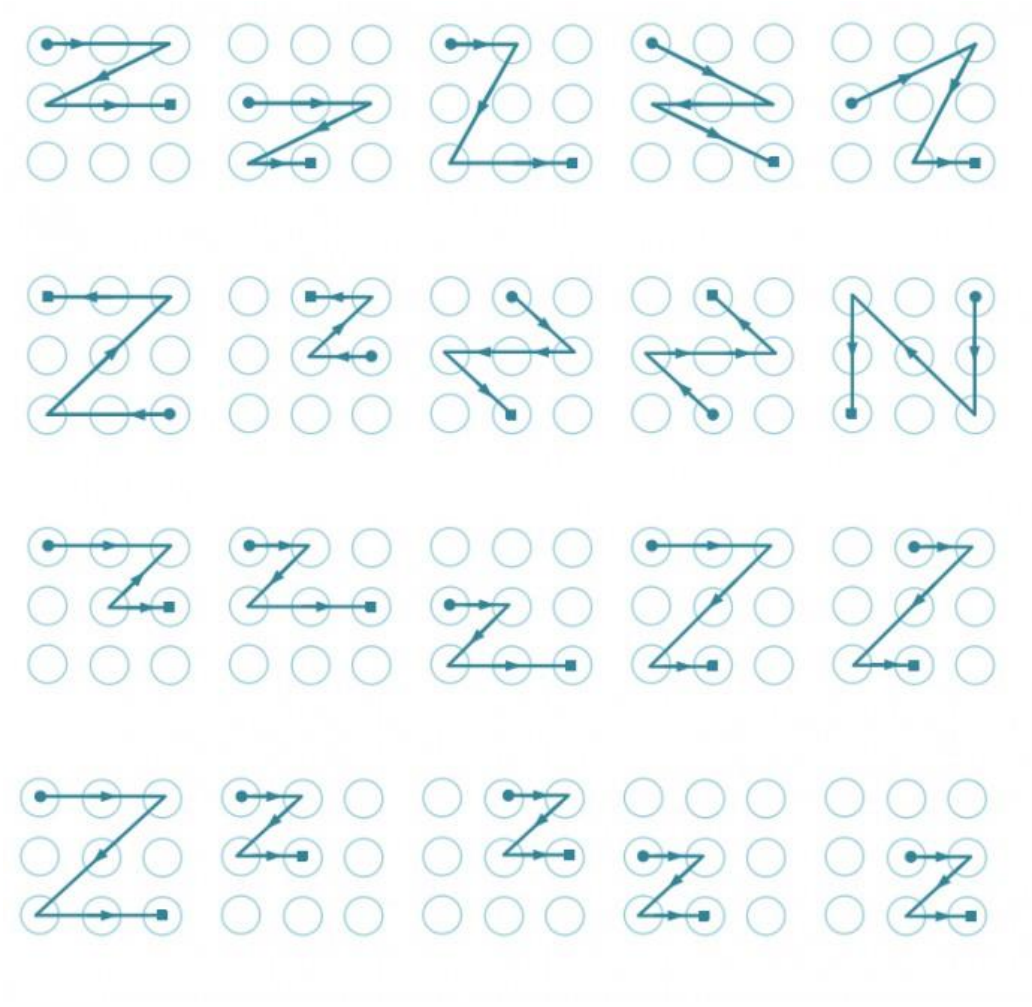


촬영(도촬) 손가락 움직임 분석 손가락 자취 > 그래프 사용자의 시선으로 변환 패턴 경우의 수 추출 사용자 잠금 패턴 획득

NDSS'17 / 2017 Internet Society

CryptoCraft LAB

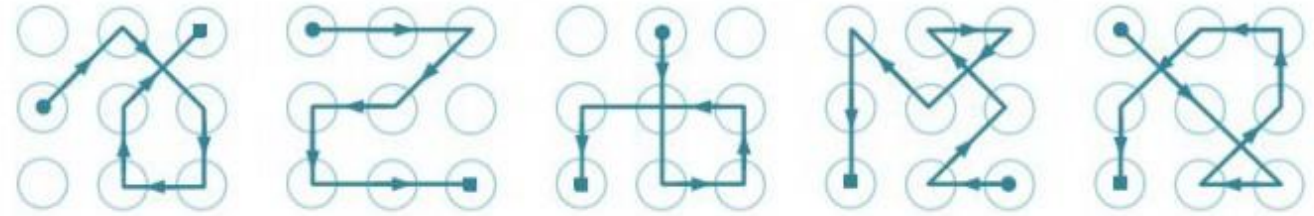
2 패턴 잠금



CryptoCraft LAB

2 패턴 잠금

(1) 교차점이 하나도 없는 가장 간단한 패턴 (예)



(2) 교차점이 1~2개인 보통 수준의 패턴 (예)

(3) 교차점이 3개 이상인 복잡한 패턴 (예)

CryptoCraft LAB

3 포인트와 진동을 활용한 패턴 잠금

포인트별로 랜덤하게 진동을 출력하는 패턴 형식 제안

기존의 패턴과 달라진 점은 포인트별로 출력되는 진동의 세기를 다시 입력해야하는 방식

기존의 포인트는 단순 포인트 기능이지만 제안 패턴의 경우 강, 중, 약으로 나누어 각 포인트 별 3^n만큼 보안 강도 향상

CryptoCraft LAB

3 제안 패턴

- 동작 과정
  1. 사용자는 기존 키패드와 같이 드래그를 진행
     - 각 포인트별로 랜덤하게 진동의 세기가 출력 됨

  2. 사용자는 진동을 통해 포인트별 강, 중, 약 진동의 세기를 파악
     - 오로지 사용자만 느낄 수 있으며, 진동 세기에 따라 같은 패턴이라도 전혀 다른 패턴이 됨

CryptoCraft LAB

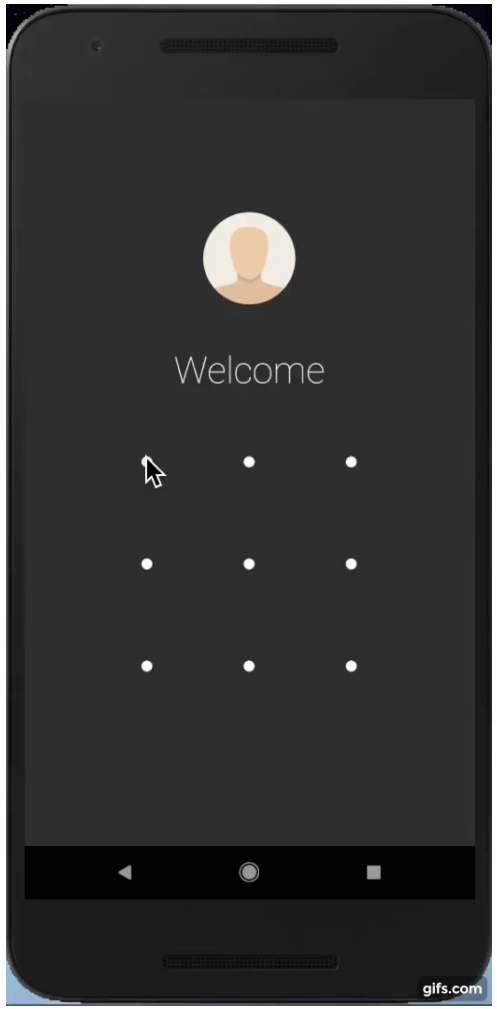
3 제안 패턴

Pseudocode for Random Vibration

```
01: void shuffle(int *arr, int size) {
02:     int tmp;
03:     int randNum;
04:     for (int i=0; i < size-1; i++) {
05:         randNum = rand() % (size - i) + i;
06:         tmp = arr[i];
07:         arr[i] = arr[randNum];
08:         arr[randNum] = temp;
09:     }
10: }
11: int main(void){
12:     int vibration[3] = {1, 2, 3};
13:     suffle(vibration, 3);
14: }
```

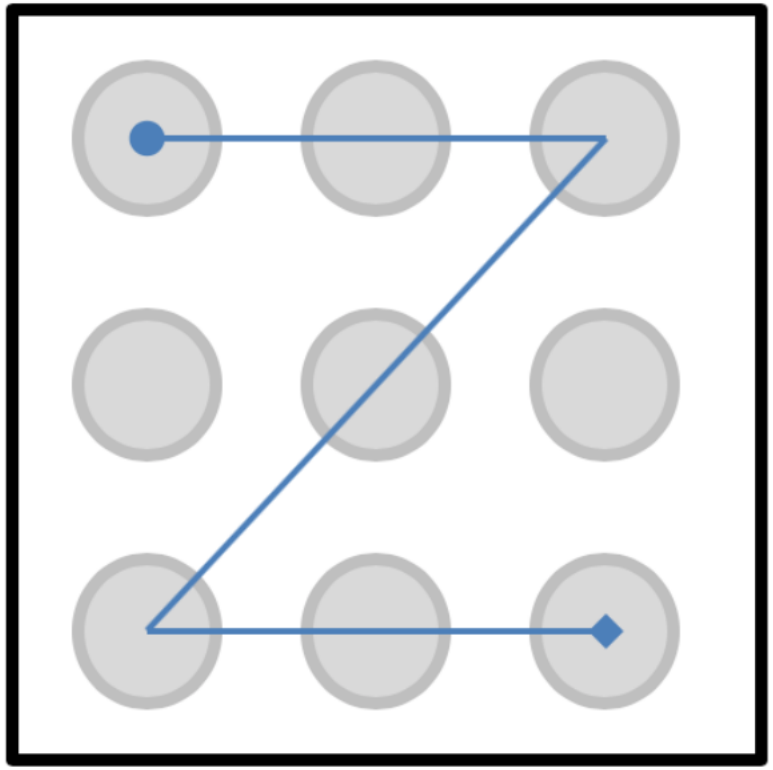
CryptoCraft LAB

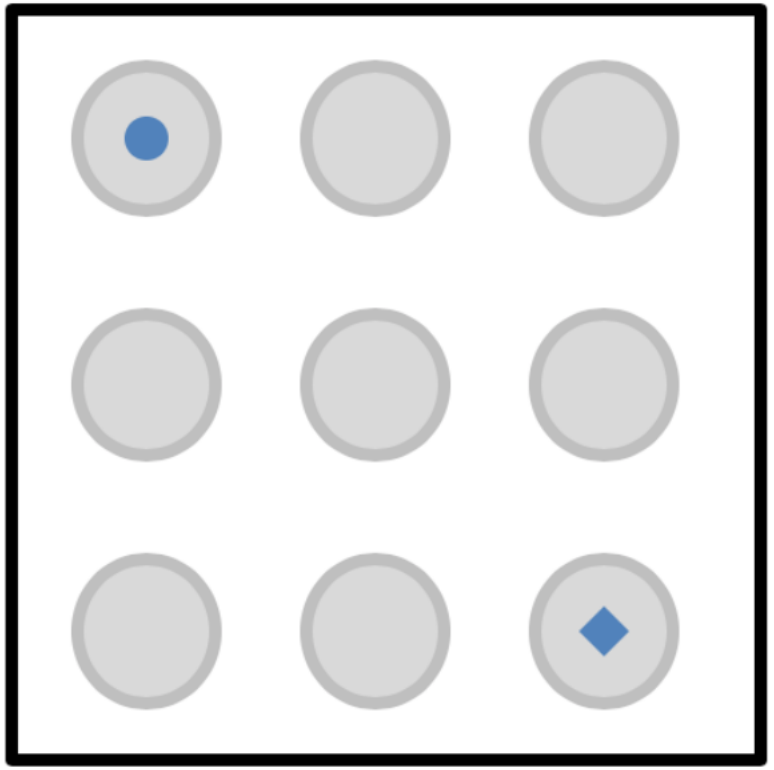
3 제안 패턴



CryptoCraft LAB

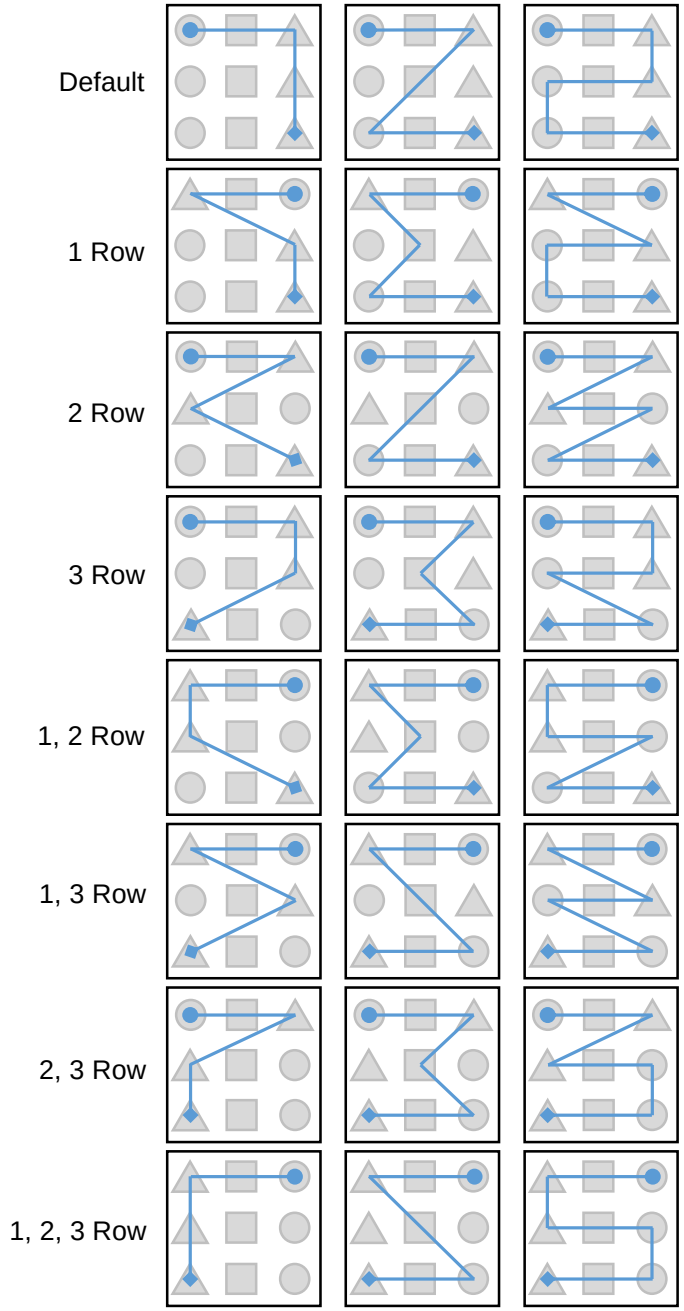
3 제안 패턴 - 비가시화



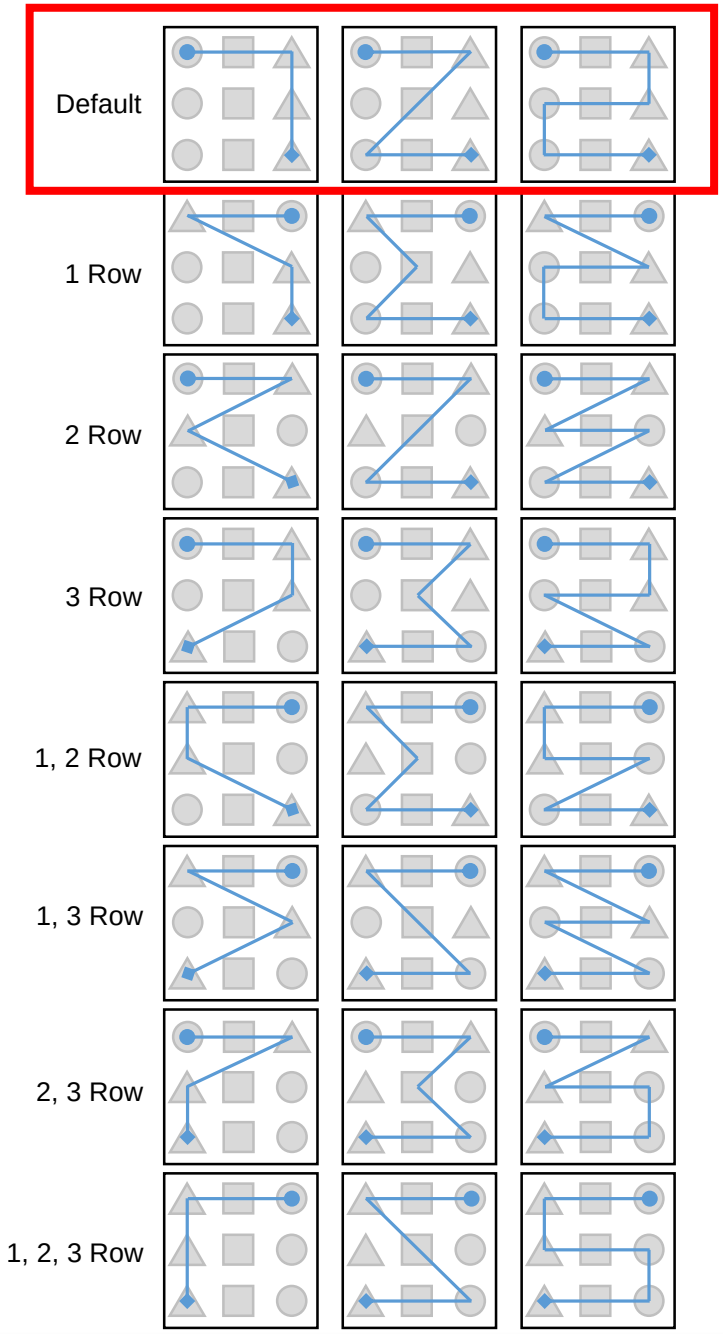


CryptoCraft LAB

행을 기준
1, 3열 바꾸기

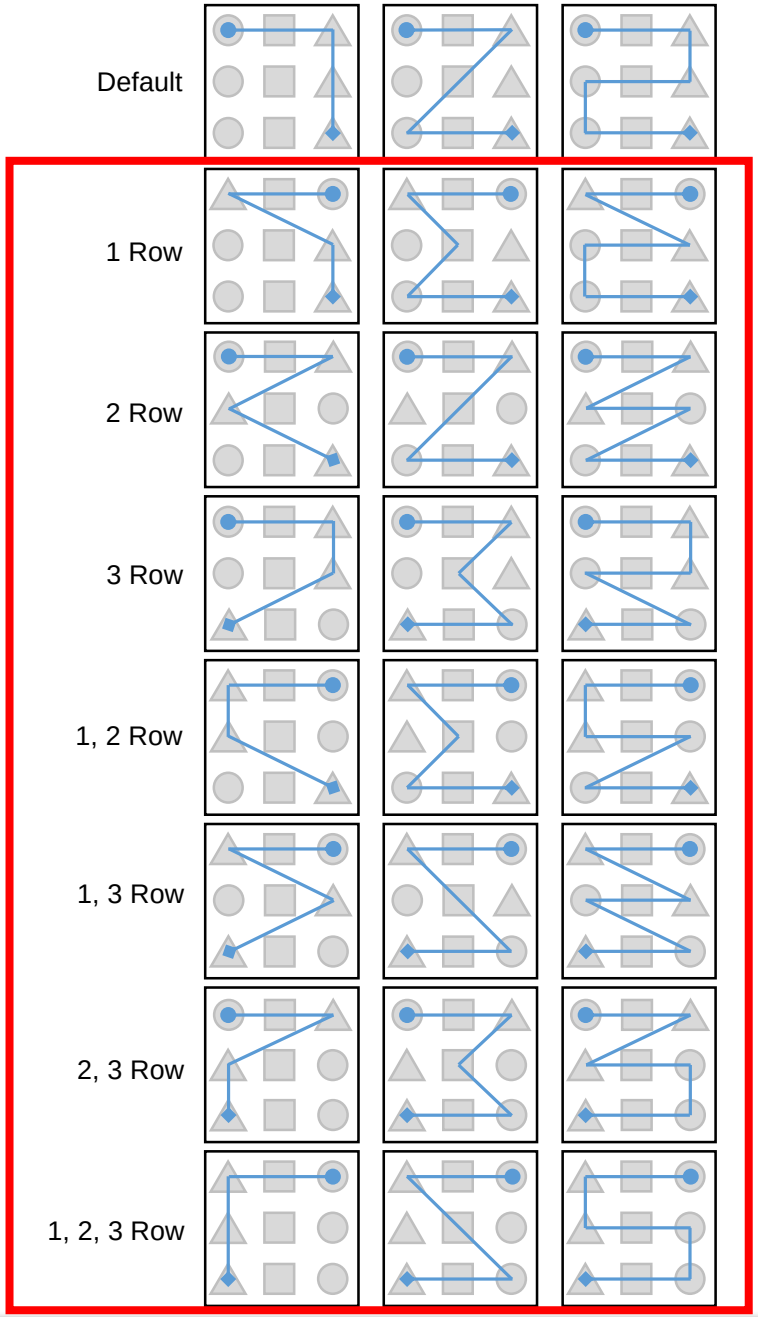


행을 기준
1, 3열 바꾸기

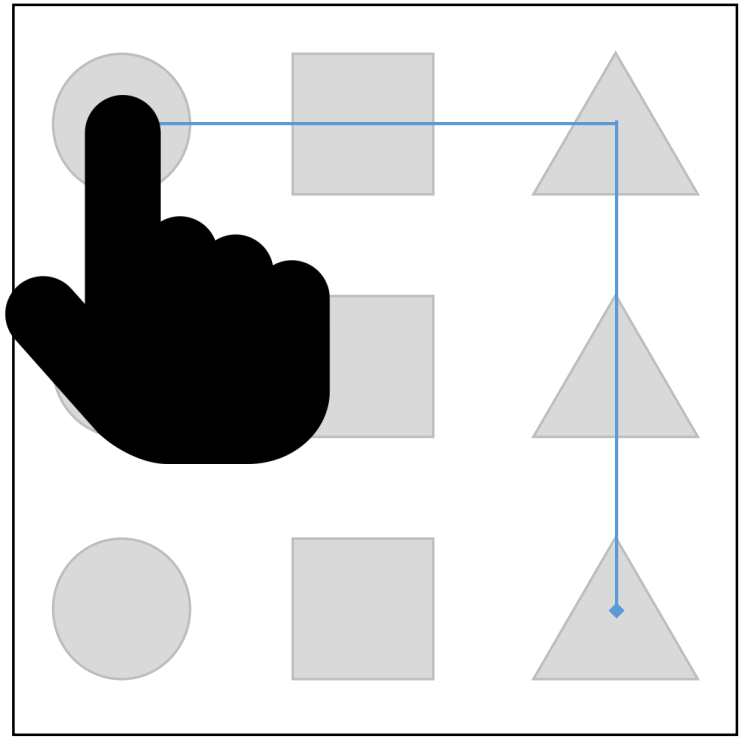


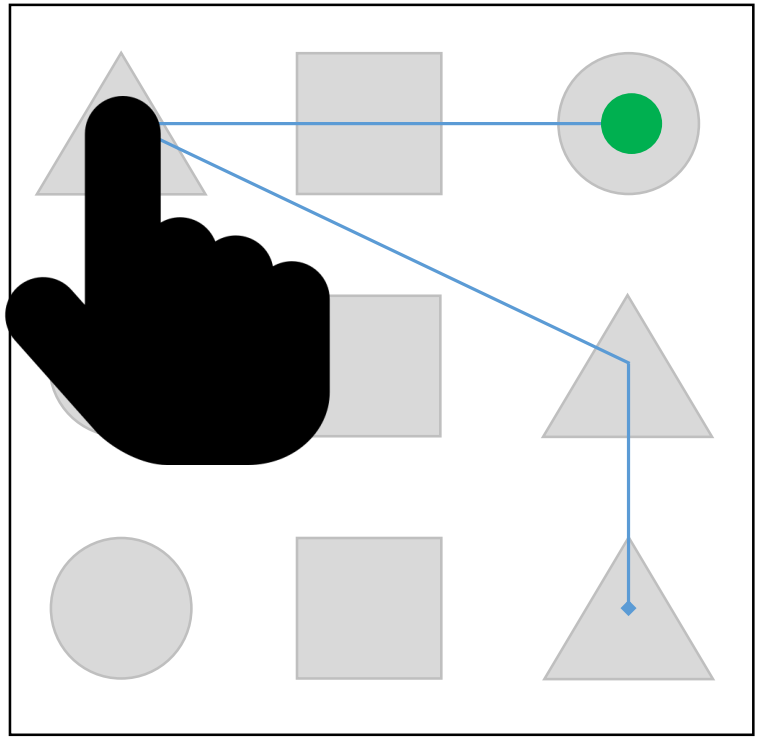
사용자가 실제 드래깅 하는 모습

행을 기준
1, 3열 바꾸기

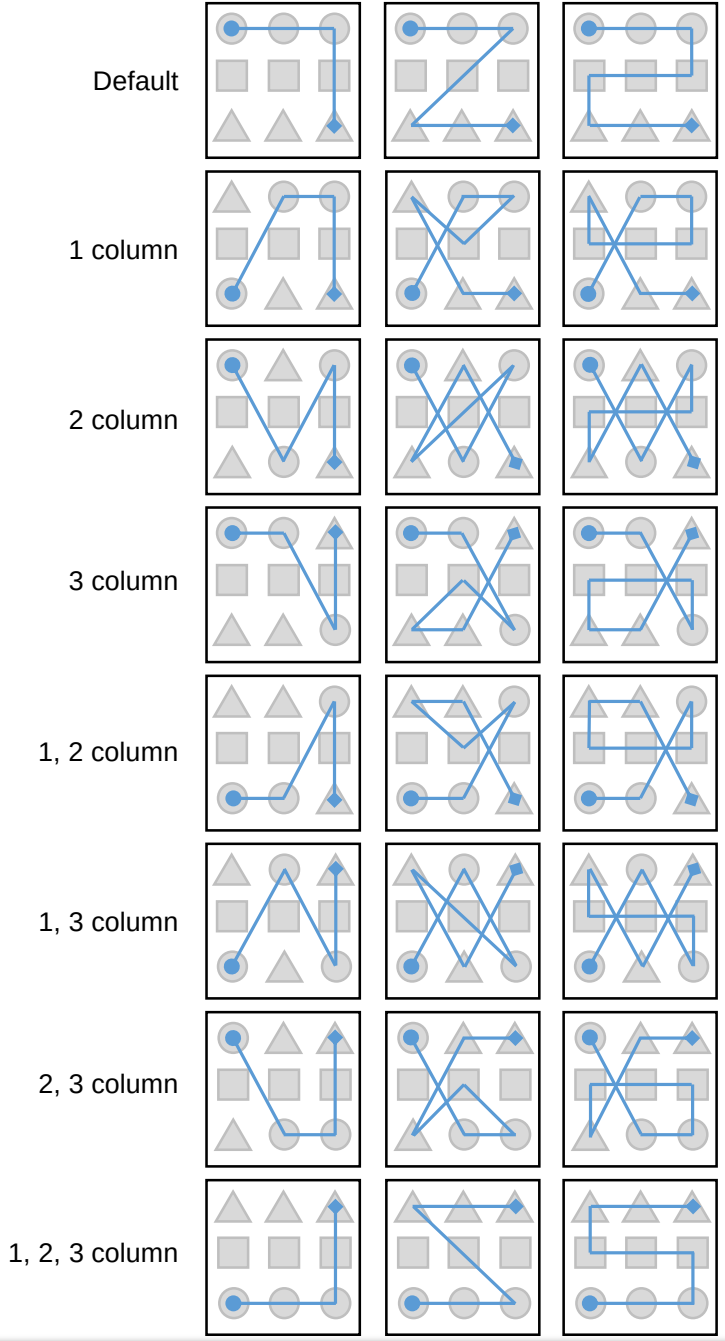


사용자 UI에 랜덤하게
보여주는 UI

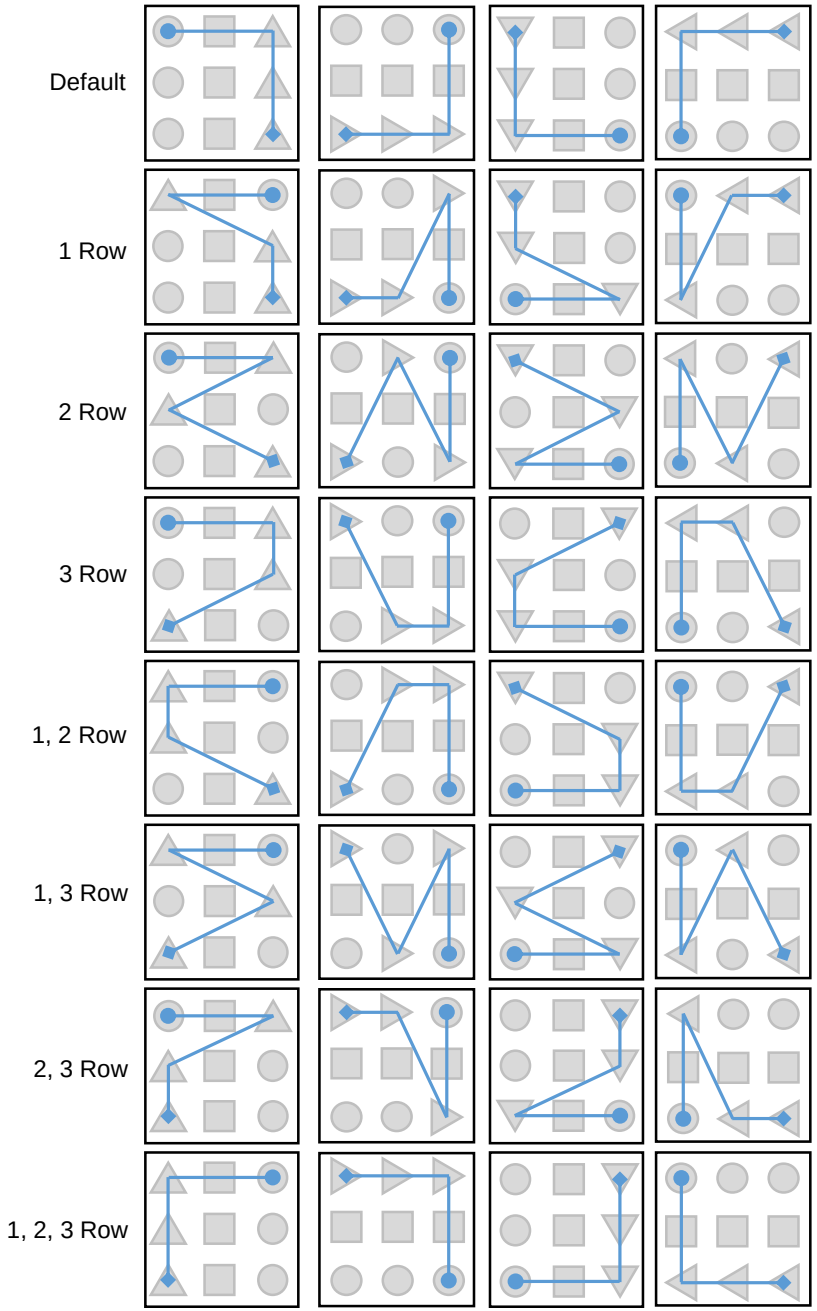




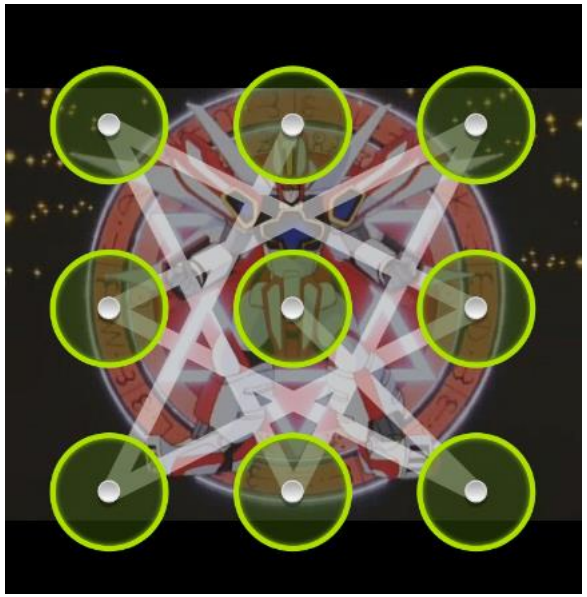
열을기준
1, 3행 바꾸기


Default
1 column
2 column
3 column
1, 2 column
1, 3 column
2, 3 column
1, 2, 3 column

열, 행을 적용한 뒤
90도, 180도, 270도 회전



4 성능 평가



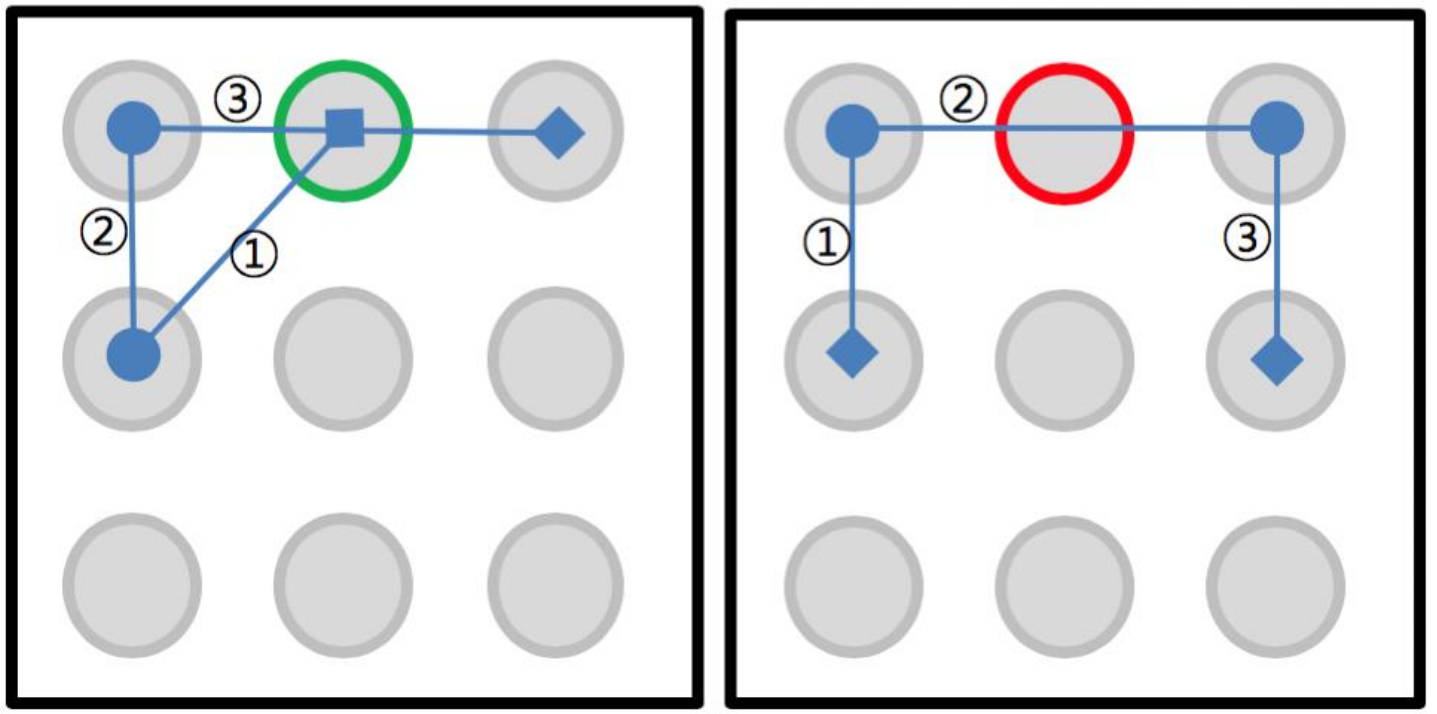
$$\sum_{4}^{9}({}_{9}C_{n} * n!)$$

n=4, 3,024
n=5, 15,120
n=6, 60,480
n=7, 181,330
n=8, 362,880
n=9, 362,880

Total = 985,824

CryptoCraft LAB

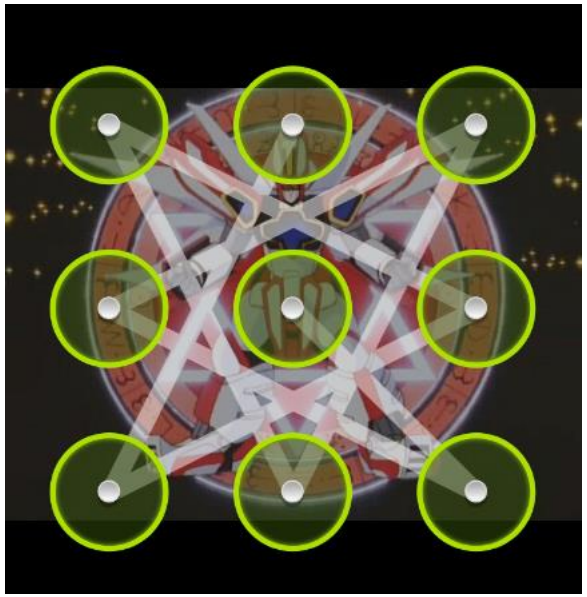
4 성능 평가



(1, 3), (3, 1), (4, 6), (6, 4), (7, 9), (9, 7), (1, 7), (7, 1), (2, 8), (8, 2), (3, 9), (9, 3), (1, 9), (9, 1), (3, 7), (7, 3)

CryptoCraft LAB

4 성능 평가

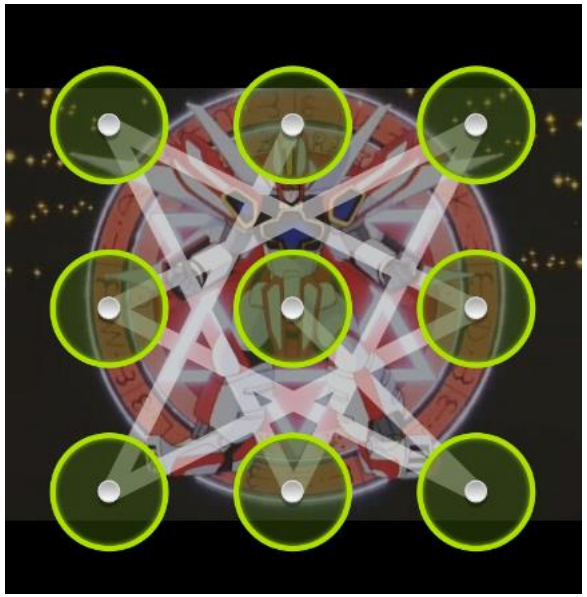


n=4, 1,624
n=5, 7,152
n=6, 26,016
n=7, 72,912
n=8, 140,704
n=9, 140,704

Total = 389,112

CryptoCraft LAB

4 성능 평가

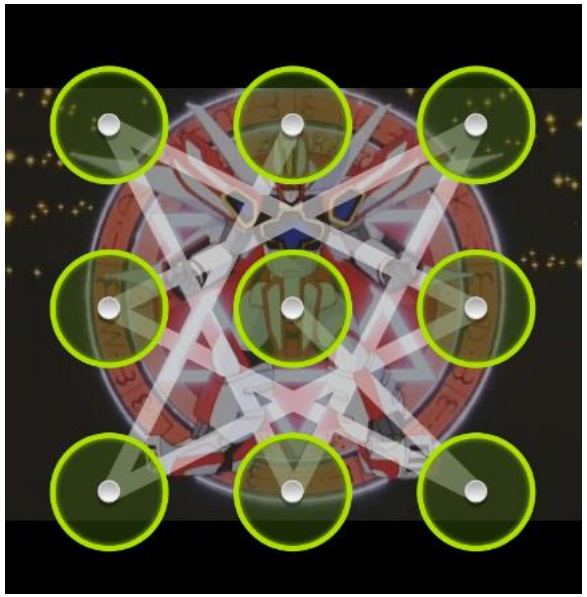


n=4, 1,624 * 3^4
n=5, 7,152 * 3^5
n=6, 26,016 * 3^6
n=7, 72,912 * 3^7
n=8, 140,704 * 3^8
n=9, 140,704 * 3^9

Total = 3,872,929,464

CryptoCraft LAB

4 성능 평가



n=4, 1,624	n=4, 1,624 * 3^4
n=5, 7,152	n=5, 7,152 * 3^5
n=6, 26,016	n=6, 26,016 * 3^6
n=7, 72,912	n=7, 72,912 * 3^7
n=8, 140,704	n=8, 140,704 * 3^8
n=9, 140,704	n=9, 140,704 * 3^9
Total = 389,112	Total = 3,872,929,464

약 9,953배 상승

CryptoCraft LAB

4 성능 평가

공격자에게 특정 패턴을 정확하게 탈취 당하였을 경우

	Default pattern lock	Suggest system
Shoulder surfing attack	1	3^n
Smudge attack	2	$3^n * 2$

CryptoCraft LAB

4 성능 평가 – 각 공격방법에 대한 안전성 비교

	Default	Visible	Invisible	Obfuscation
Brute-Force Attack	Low	Middle High	Middle High	Middle High
Shoulder Surfing Attack	Low	Middle	High	High
Recording Attack	Super Low	Middle	Middle High	Middle High
Smudge Attack	Super Low	Middle High	Middle High	Middle High

CryptoCraft LAB

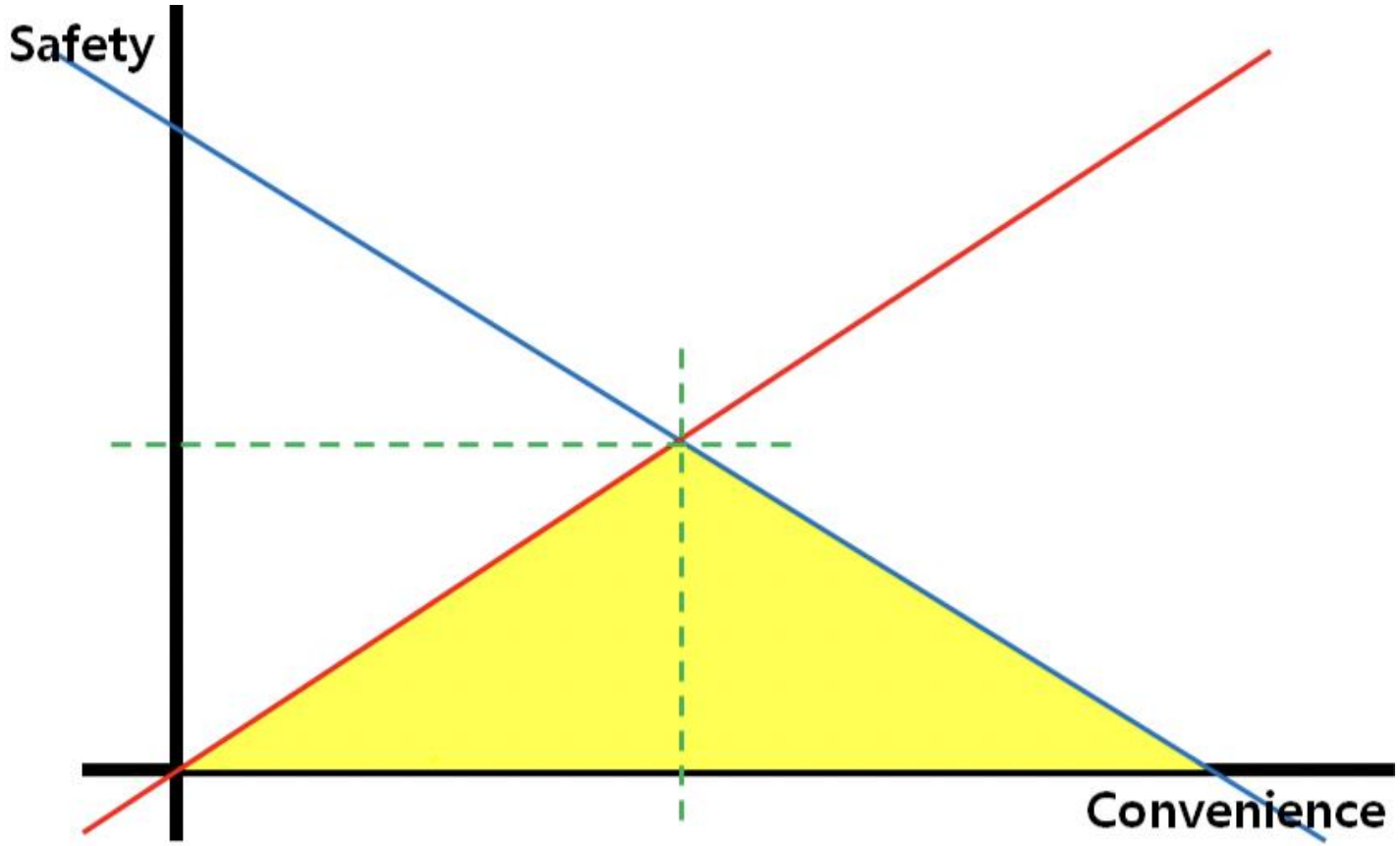
4 성능 평가 – 사용자 편의성 비교

20대 10명, 30대 10명, 40대 10명 그리고 50대 10명 총 40명의 사용자를 기반

	Default	Visible	Invisible	Obfuscation
Password Registration	Easy	Middle	Hard	Hard
Login Process	2.9s	8.4s	10.0s	10.6s
Easy to Remember	Easy	Middle-Hard	Middle-Hard	Middle-Hard
Typing Speed	2.9s	8.4s	9.7s	11.4s

CryptoCraft LAB

5 결 론



CryptoCraft LAB

5 결 론

- 기존 잠금 패턴의 총 경우의 수는 389,112번으로 전수조사하는데 큰 어려움 없음
  그러나 제안하는 잠금 패턴은 3,872,929,464번으로 약 9,953배 향상

- 기존 패턴의 경우 패턴이 노출되면 방어 대책이 없음
  - 입력시간을 측정하는 쓰레드는 패턴이 노출되어도 입력 진동에 따라 다른 패턴이 되기 때문에 각 포인트별 3^n만큼 보안 강도를 높힘

- 보안 필름 같이 별도의 장비 없이 보안 강도를 향상 시킬 수 있음

- 제안한 패턴은 기존보다 편리성은 떨어지나 강력한 보안성을 제공

CryptoCraft LAB

참고

UCSIS. “Shoulder Surfing attack in graphical password authentication,” Available: https://arxiv.org/ftp/arxiv/papers/0912/0912.0951.pdf.

A. J. Aviv, K.L. Gibson, E. Mossop, J. M. Smith, “Smudge Attacks on Smart phone Touch Screens,” Woot, 10: 1-7, 2010.

C. Dongmin, “Application Adaptive Pattern-based Authentication Method for Smartphones,” Asia-pacific Journal of Multimedia Services Convergent with Art, Humanities, and Sociology Vol.8, No. 2, pp. 59-67, February. 2018.

T. Kwon, S. Na, “TinyLock: Affordable defense against smudge attacks on smartphone pattern lock systems.” computers & security, 42, pp. 137-150. 2014.

Youtube. “The video of how it works,” Available: https://youtu.be/OEOkHHQPTgA.

Github, “The open source of press time pattern lock PIN” Available: https://github.com/kyu-h/PressTime_PatternLock_PIN.

A. Karawash, “Brute Force Attack,” Available: https://www.researchgate.net/profile/Ahmad_Karawash/publication/299645572_Data_protection_and_Brute_Force_attack/links/5703c19e08aeade57a25ae7b/Data-protection-and-Brute-Force-attack.pdf.

CryptoCraft LAB

Thank You