ML HACKING

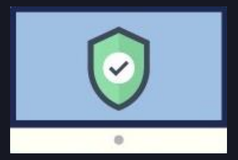
THREATS EVOLVING INTO MACHINE LEARNING

IGLOO SECURITY

Why do we need ML ?

Traditional Attack

- 악의적인 공격자들의 공격 패턴이 다양하지 않음
  - Web : SQL Injection, XSS, CSRF, Parameter 변조 등
  - Malware : 파밍 악성코드, Bot, RAT, Trojan Horse 등
    자동화된 악성코드 제작툴 활용
  - DoS, DDoS, DRDoS, Backdoor, Hijacking ...
- 기존 방식의 보안 솔루션들로 공격을 효과적으로 방어



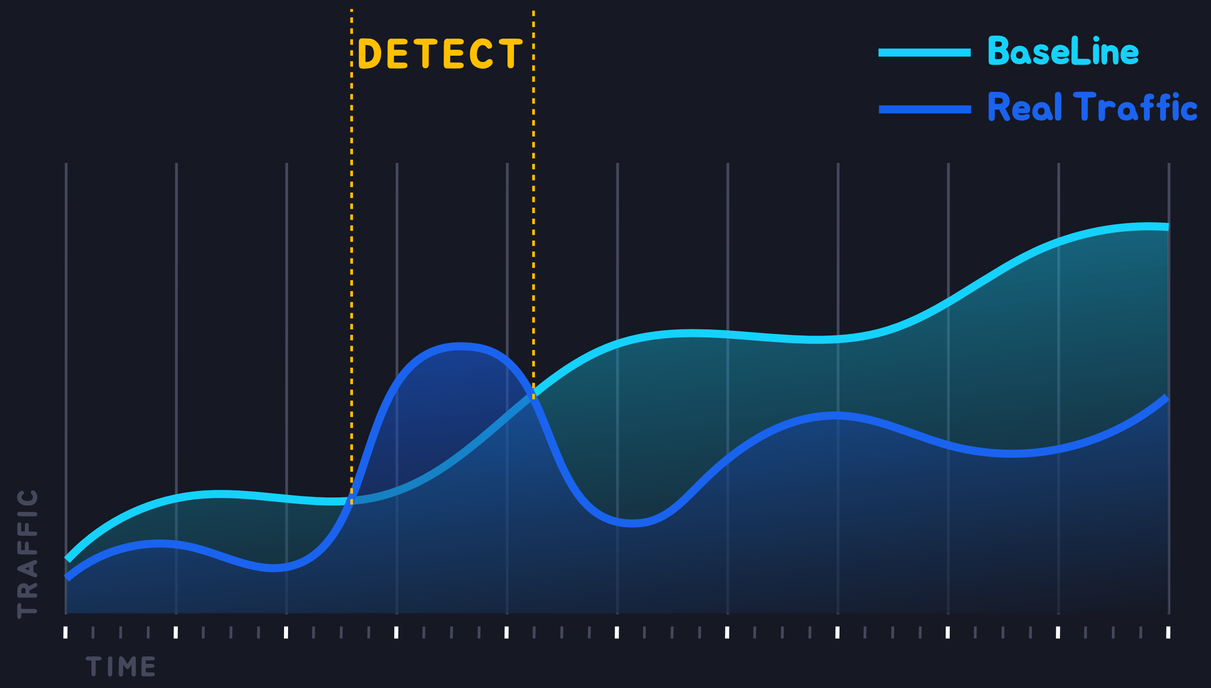
New Attack

- 넘쳐나는 수많은 데이터의 양
- 예측할 수 없는 공격자들의
  다양한 변종 공격 패턴

Network Intrusion Detection

- 대부분의 네트워크 침입 탐지는 네트워크 트래픽을 기반으로 하는 비정상 행위 탐지 방식을 사용
  (Network Anomaly)
- 알려진 공격 : IDS/IPS + e.t.c.
  하지만, 알려지지 않은 공격은...?

1. Network Traffic Collection
2. Gathering Feature
   - Header Info, IP Address, Traffic Length...
3. Feature Analysis
   - Out of Baseline Traffic
4. Anomaly Traffic Detect



Malware Analysis

- 복잡한 난독화 알고리즘과
  정교한 구조로 제작된 악성코드

〔목표〕

정보 유출, 시스템 파괴, 악의적인 다운로드, 좀비 등 ...

- SandBox + Malware + ML
  - 샌드박스 환경 구축 후 악성코드 실행
  - 악성코드 동작으로 인한 행위 정보 수집
    (ex. API Call / Parameter...)
  - 특징 추출 (Clustering / Classification)

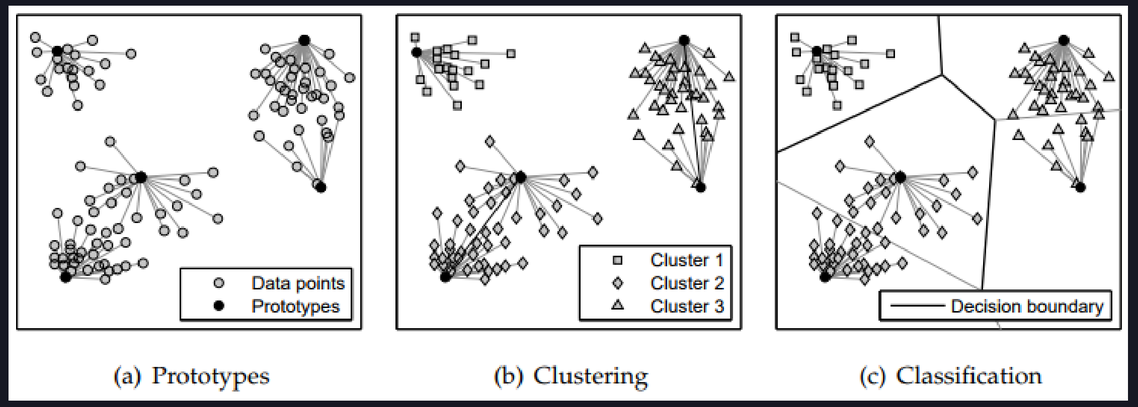
Malware Analysis

- 특정 System Call 의 호출을 탐지
- 탐지된 System Call 을 통해 작업 수행 여부 파악
- 파악된 정보를 바탕으로 벡터 공간에 출력
- "군집화" 와 "분류" 기법을 이용하여 특징을 분석
  (Clustering / Classification)

[Advantage]

분석할 파일이 일반적인 악성코드 행위 유형과 비교

➡ 악성코드의 특성이나 유형을 식별



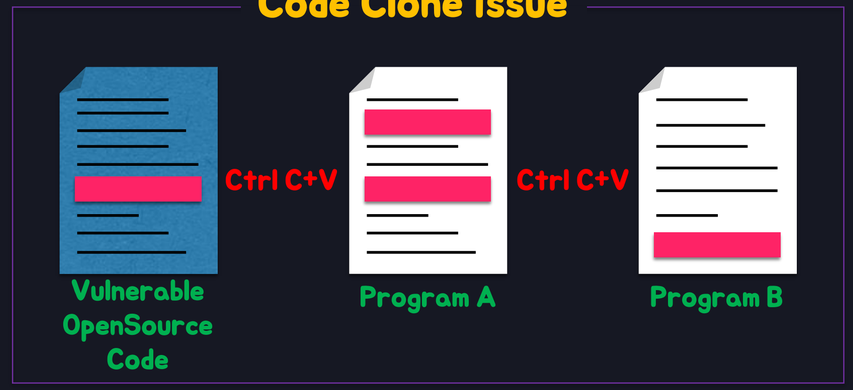
Software Vulnerability Analysis

- NVD 또는 Exploit-DB 같은 공개 데이터 베이스의 정보를 활용하여 취약점 분석 수행

This code snippet deserializes an object from a file and uses it as a UI button:

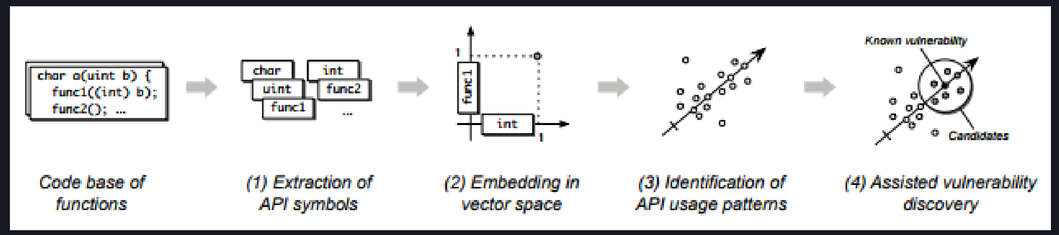
Example Language: **Java**

```
try {
  File file = new File("object.obj");
  ObjectInputStream in = new ObjectInputStream(new FileInputStream(file));
  javax.swing.JButton button = (javax.swing.JButton) in.readObject();
  in.close();
```



(Syntax Tree)

- 구문 트리의 활용을 통해 프로그래밍 패턴을 식별하여 머신러닝 학습 및 분석으로 새로운 취약점 발견



[Disadvantage]

- ⊗ 취약점과 관련된 데이터의 양이 충분하지 않음
- ⊗ 컴파일된 바이너리 파일 분석 시 많은 변수가 존재

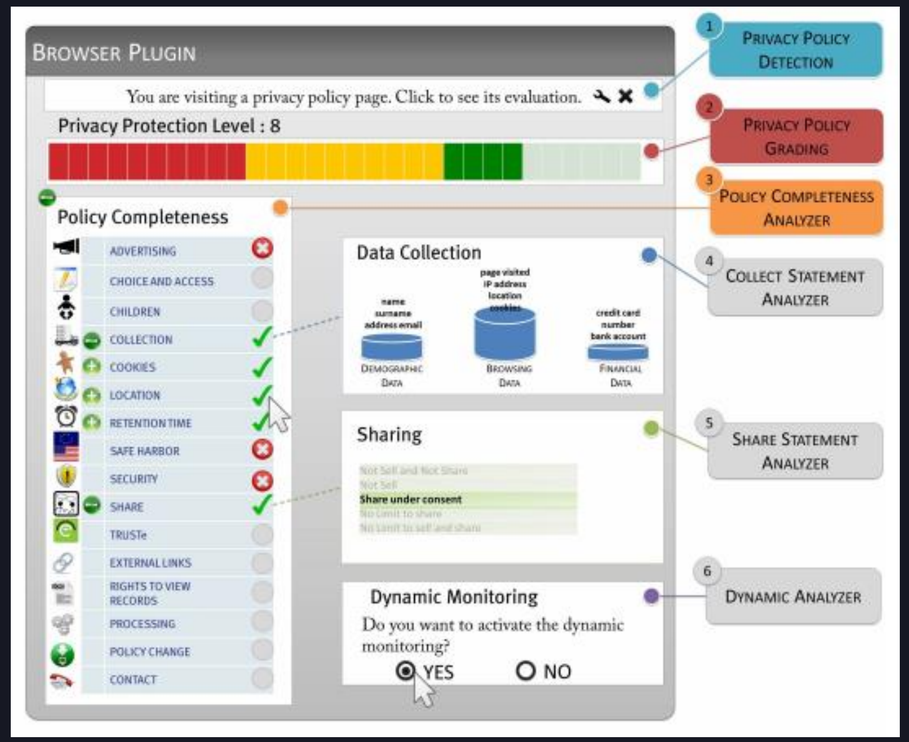
  (ex. Programming Language, Coding Style, Compiler Version, ...)

Compliance

- 개인정보 사용 동의에 관한 문장이나 단어 등을 분석
- 특정 체크리스트 포맷을 만들어 사용자가 이해하기 쉽게 표현

Blah Blah Blah ..... Pass

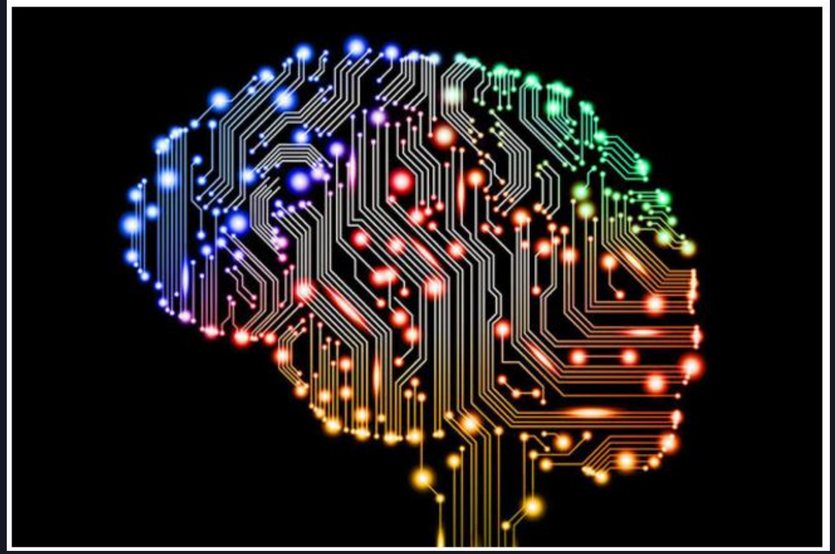
Advertising .... Check
Safe Harbor ... Check
Security ..... Check
...



보안 솔루션은 머신러닝으로 진화 중… 최신 사례 4가지

Maria Korolov | CSO

머신러닝의 발전으로 보안 시스템을 더 쉽게 훈련시킬 수 있게 된다. 또 변화에 더 유연하게 대응할 수 있게 된다. 머신러닝이 보안 분야에 적용되고 있는 최신 사례들을 정리했다.



보안업계, 머신러닝 기법 적용 활발…기계적 판단으로 허점 차단

기자] 보안업계가 머신러닝 기법을 도입한 제품 및 서비스 출시에 적극 나서고 있다.

최근 사람의 심리를 이용하는 '사회공학적 이메일 해킹' 기법 등 피해를 입는 기업들이 나타나면서 이에 대응할 수 있는 보안 솔루션 출시가 이어지고 있다.

사이버전은 속도전, 보안업계도 '머신러닝' 열풍

네트워크에서 엔드포인트에 이르는 모든 영역에서 보안 위협이 고도화됨에 따라 머신러닝(기계학습)을 보안 솔루션에 접목하고자 하는 시도가 주목받고 있다. 축적된 보안 인텔리전스를 기반으로 인공지능 대응 체계를 마련, 자동화된 방식으로 보다 신속하게 위협을 탐지하고 대응하기 위한 것이다.

▲최근 주목받고 있는 머신러닝과 인공지능은 보안 분야에서도 뜨거운 화두 중 하나다.(사진= MS)

'머신러닝'으로 보안도 진화 중

국내외 보안 기업 머신러닝 적용에 '잰걸음'

2016년 11월 29일 오전 06:00

기자] 보안 업계에 머신러닝 바람이 불고 있다. 최근 글로벌 보안 업체들은 새로운 악성코드, 알려지지 않은 위협을 감지하기 위해 보안 제품에 머신러닝을 적극 도입하고 있다.

IGLOOSECURITY

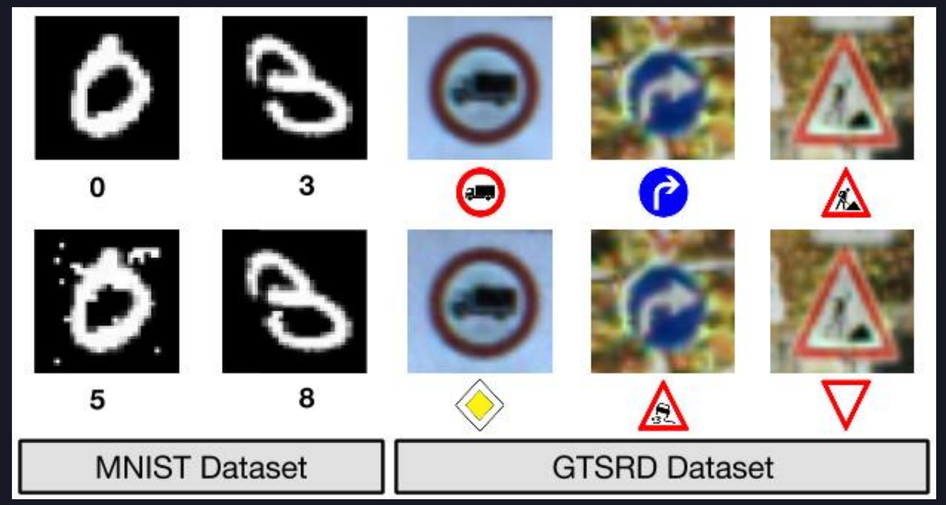
MACHINE LEARNING HACKING

THREATS EVOLVING INTO MACHINE LEARNING

Traditional Attack Automation

- ☑ Attack Automation
- ☑ Increased Attack Speed
- ☑ Increased Thread

Machine Learning Vulnerability #1



Overfitting

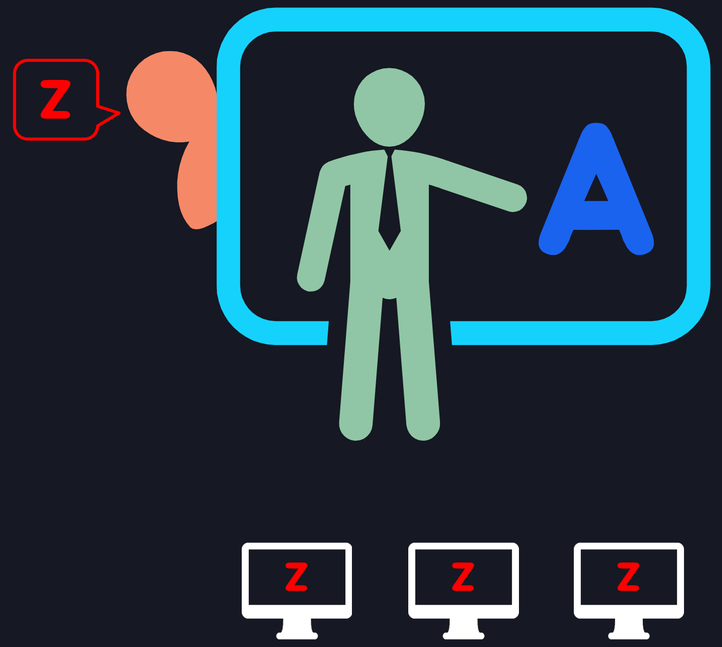
(과적합)

- [목적] : 트레이닝 데이터에는 없는 새로운 데이터를 정확하게 예측
- ⊗ 트레이닝 데이터에 지나치게 맞춰진 모델은 (Over-fit)
  새로운 데이터 예측에 실패할 가능성이 존재
- Regularization , Intelligent Learning Data , SVM ...

Cat !

?

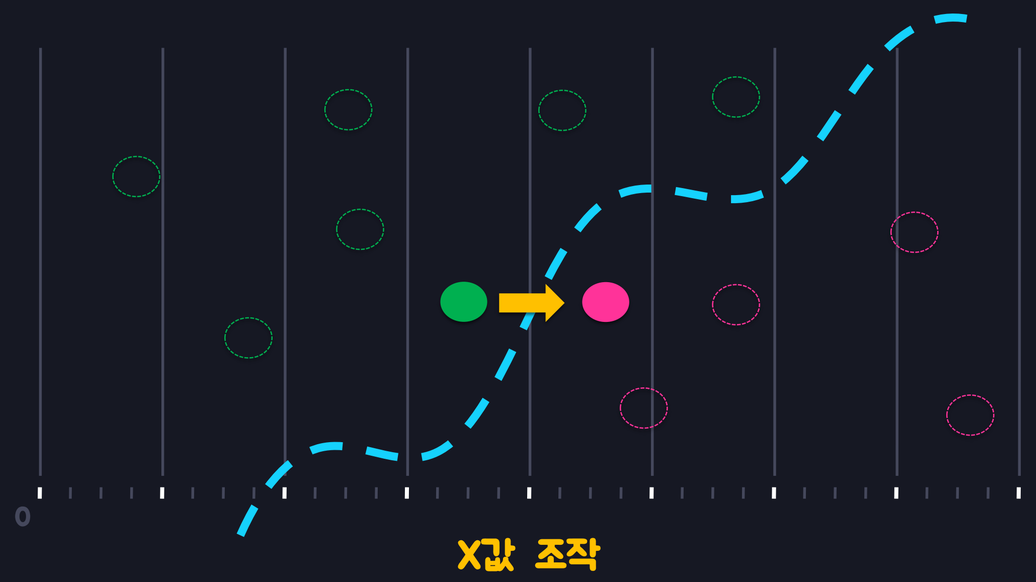
Machine Learning Vulnerability #2



- 모델의 학습 알고리즘의 취약점을 악용
- 개발자의 의도와 다르게 학습을 시켜서 공격을 수행
  (공격자의 의도대로 학습을 시킴)

마이크로소프트(MS)가 사람과 대화를 나누는 인공지능(AI) 채팅봇 '테이'(Tay)를 선보였다가 16시간만에 운영을 중단했다.

백인 우월주의자와 여성·무슬림 혐오자 등이 모이는 익명 인터넷 게시판 '폴'(boards.4chan.org/pol/)의 사용자들이 테이를 '세뇌'시켜 욕설, 인종·성차별 발언, 자극적인 정치적 발언 등을 하도록 유도한 탓이다.



Machine Learning Vulnerability #2

Create Hostile Cases

(적대적 사례 제작)

- 의도적으로 데이터 조각을 조작해서 기계학습 모델이 분류를 잘못하도록 유도
- 설계 구조가 다른 신경망이라도 유사한 데이터로 훈련되었을 경우 취약

Machine Learning Vulnerability #2

i. 신경망을 속일 사진을 제공 (Feed)

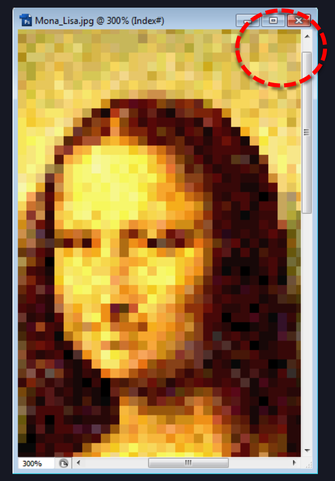
ii. 신경망의 예측 결과 및 "원하는 답변"과의 차이 확인

iii. BackPropagation으로 사진을 Pixel 단위로 조작
("원하는 답변"에 가까워 지도록)

iv. 신경망의 결과가 잘못될 때까지 위 단계를 반복

Create Hostile Cases

(적대적 사례 제작)



Monalisa.jpg

Machine Learning Vulnerability #2

Black-box Attack

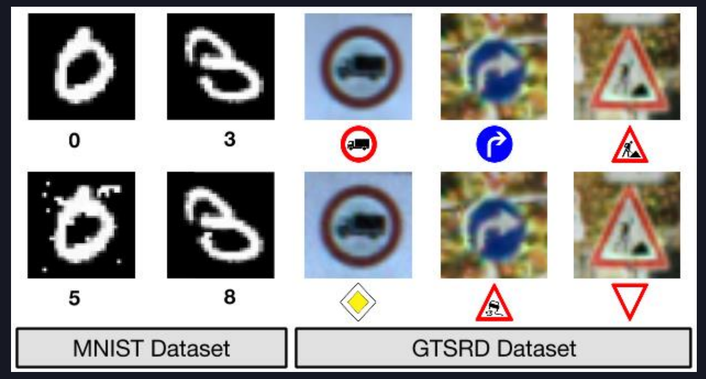
Process

i. 타겟 신경망을 속이기 위한 '훈련'이 필요
(타겟 신경망의 복사본이 필요)

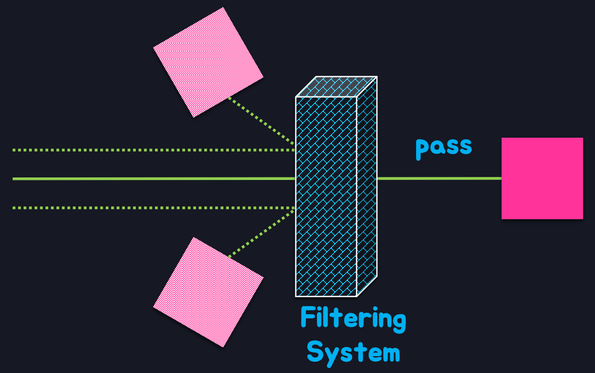
ii. 타겟 신경망의 코드는 구할 수 없음 Attack Fail?

iii. 타겟 신경망의 동작을 조사하면서 공격자는 *대체 신경망*을 구성

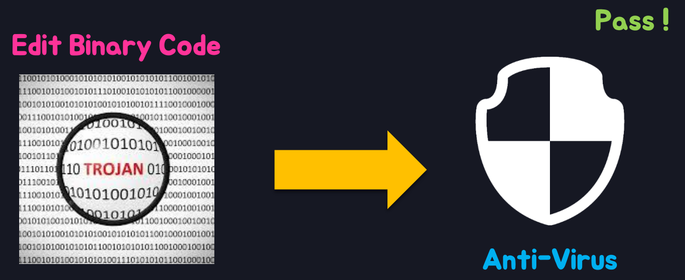
iv. 공격자의 *대체 신경망*이 타겟 신경망을 미러링하도록 훈련



Autonomous Driving



Bypass Content Filtering



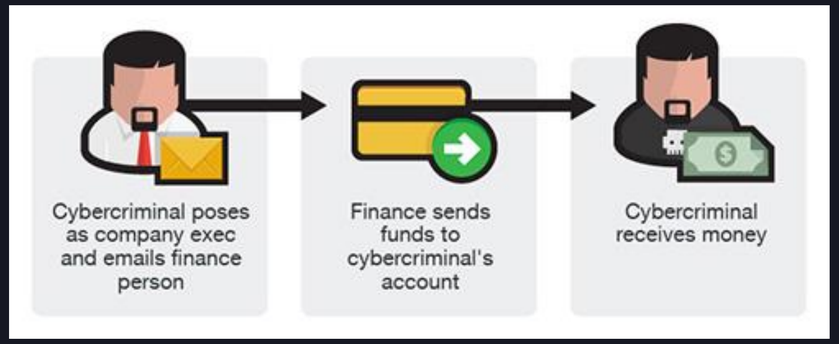
Generate Malicious Code and Bypass AV

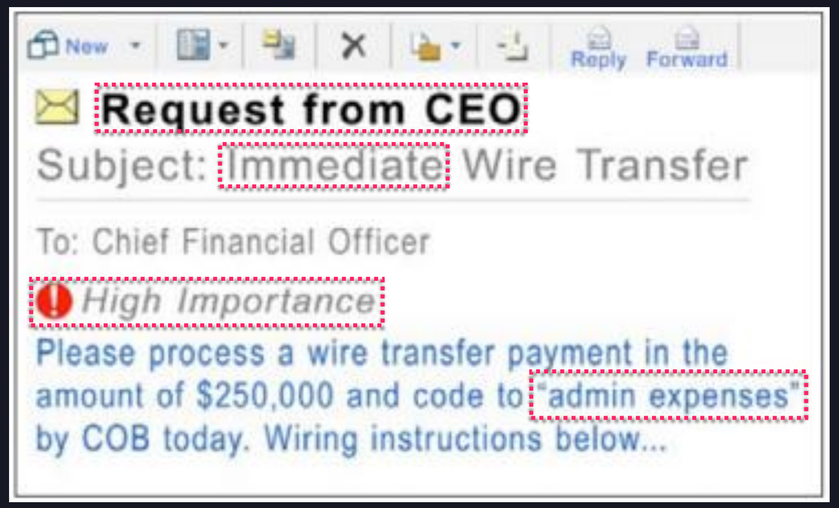
ML Hacking Case #1

Business

Email

Compromise





Select words with a High probability of Success

Request from CEO

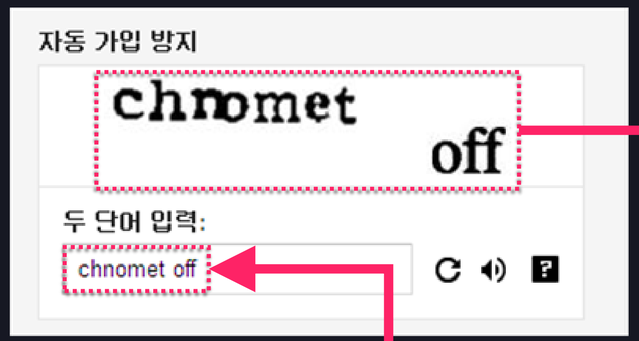
High Importance

"admin expenses"

Immediate

ML Hacking Case #2

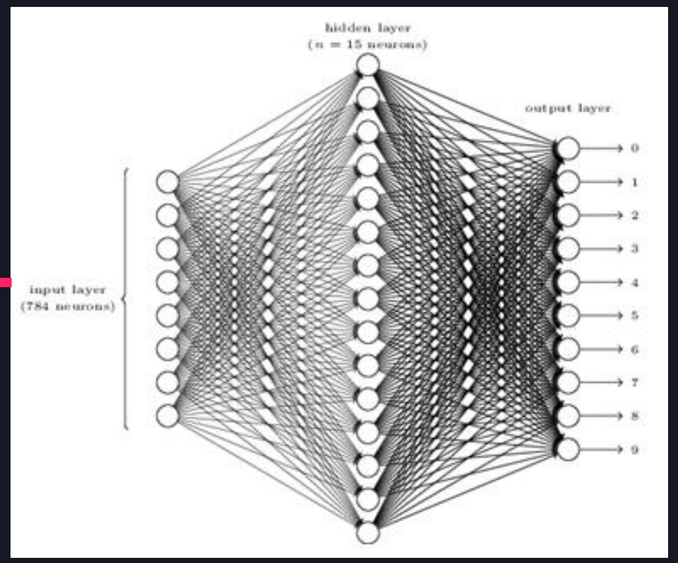
Captcha : Prevent automatic website signup



Recognition

c h n o m e t o f f

Processing



Auto Fill

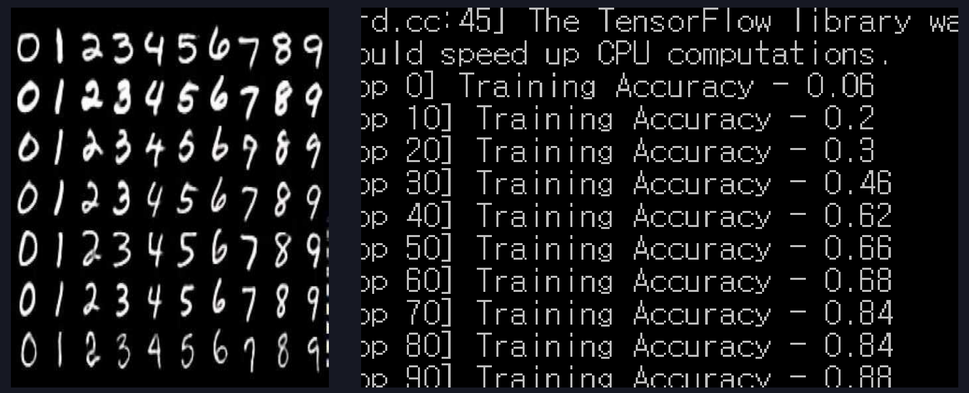
Bypass

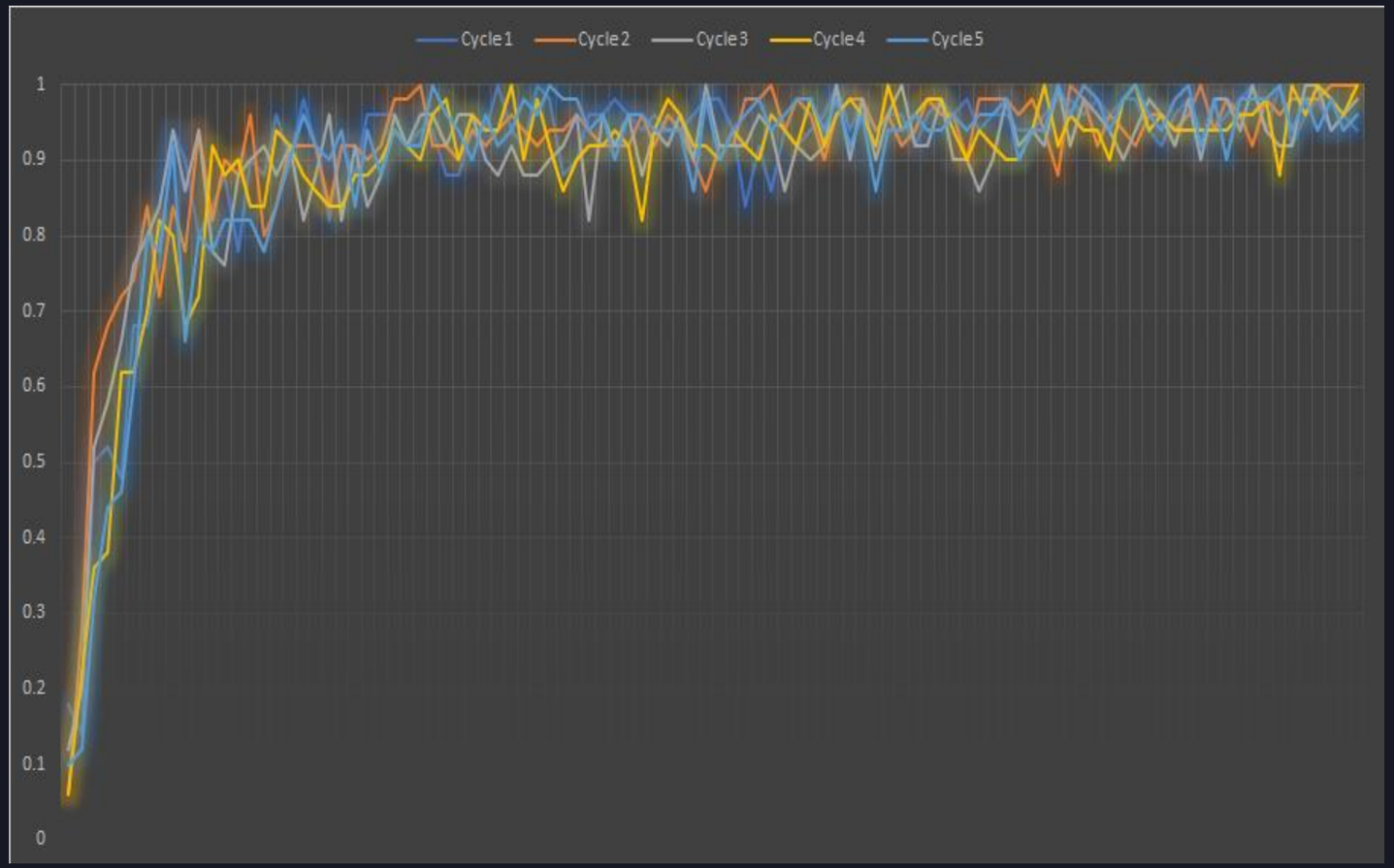
ML Hacking Case #2

Convolutional Neural Network

MNIST with CNN model

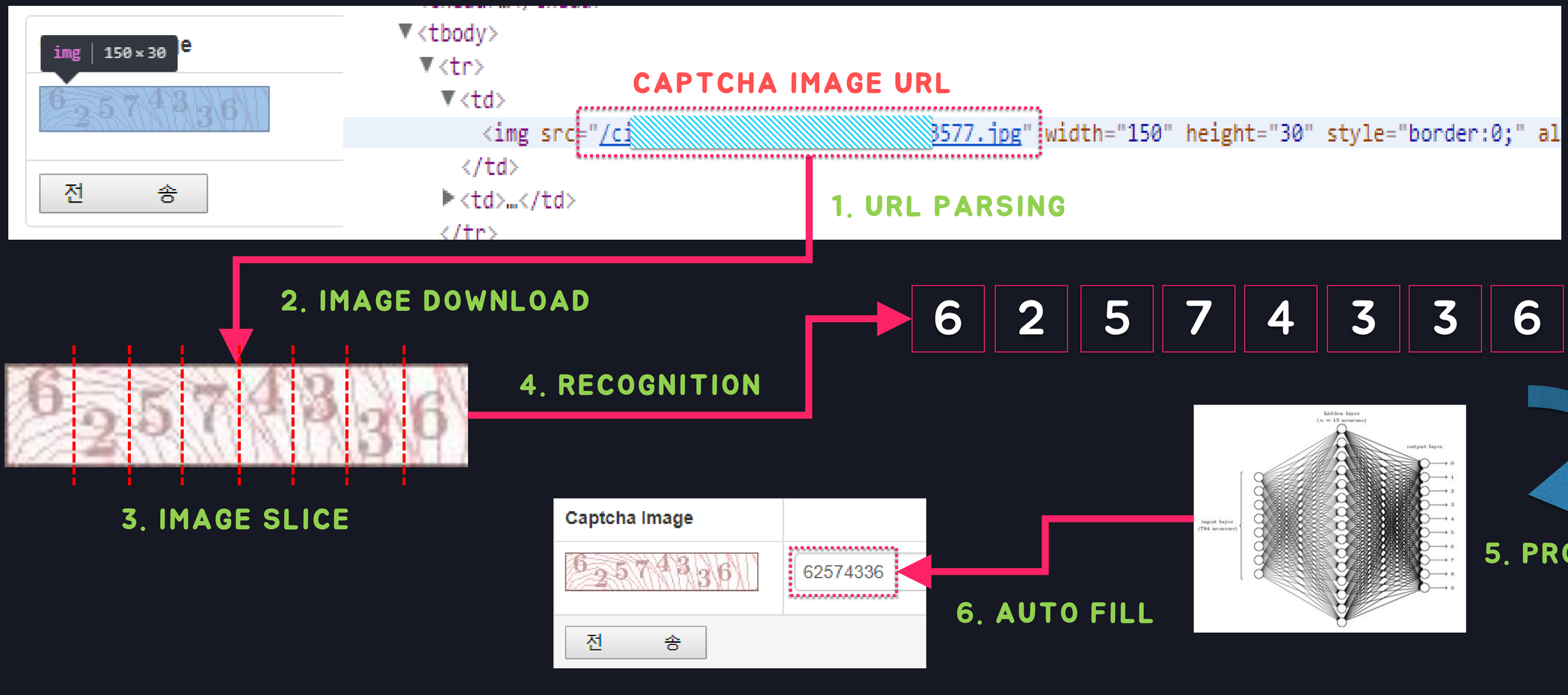
TRAINING DATA	55,000
TEST DATA	10,000
VALIDATION DATA	5,000





ML Hacking Case #2

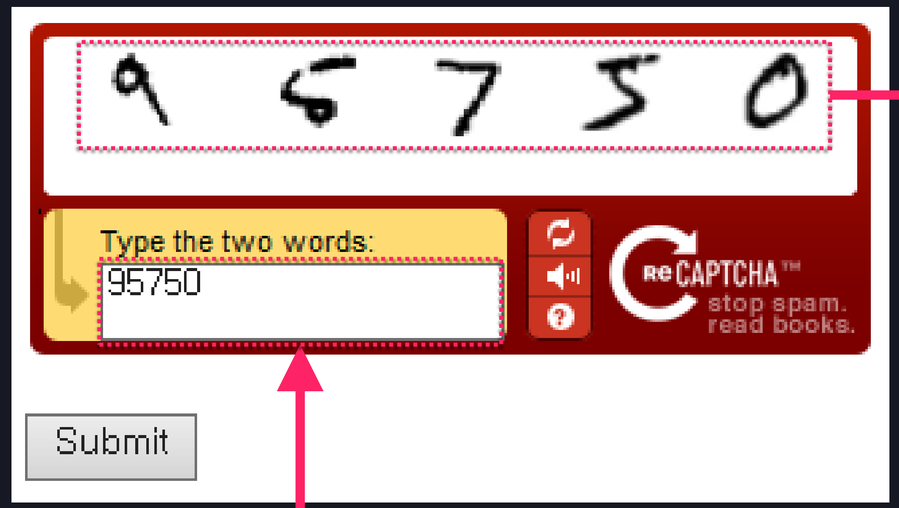
FOR EXAMPLE ...



ML Hacking Case #2

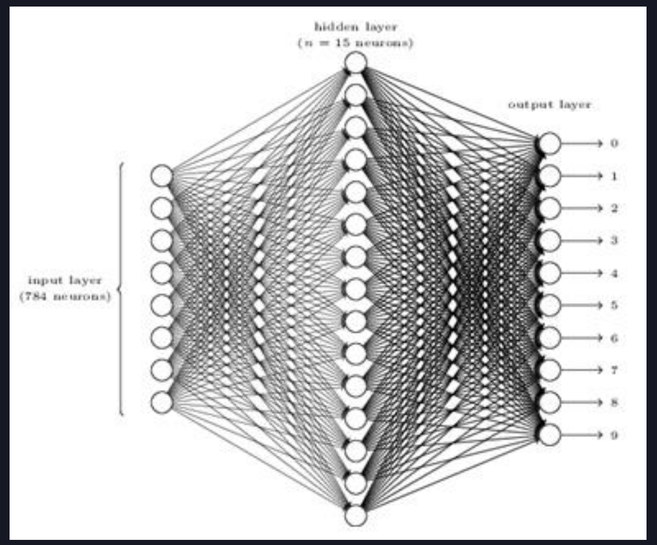
TEST !

1. IMAGE DOWNLOAD & SLICE

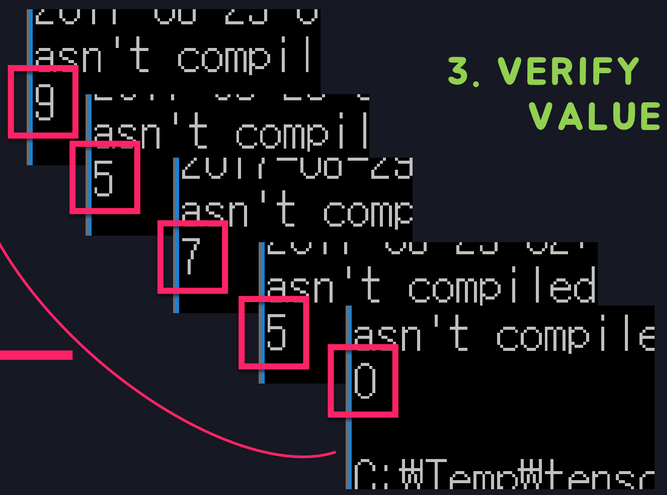


9 5 7 5 0

2. PROCESSING



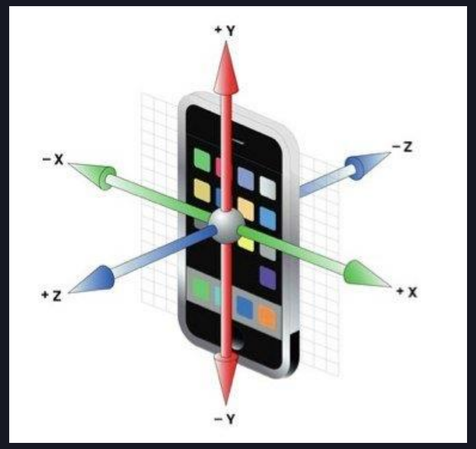
3. VERIFY VALUE

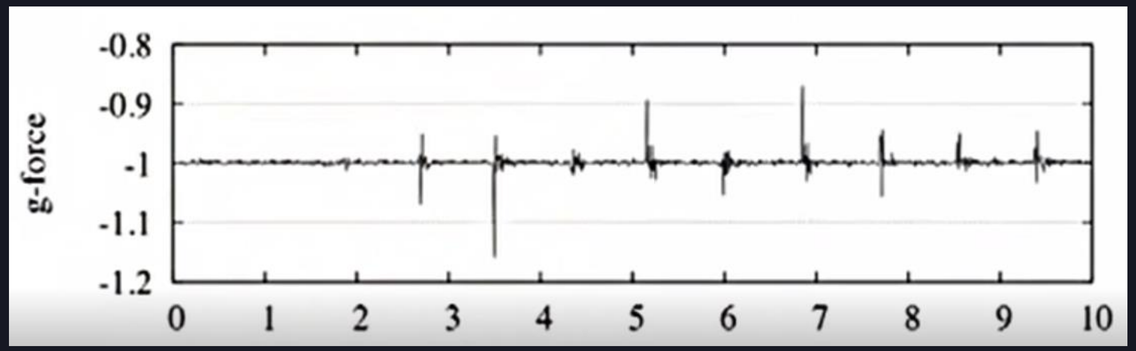


4. AUTO FILL

ML Hacking Case #3

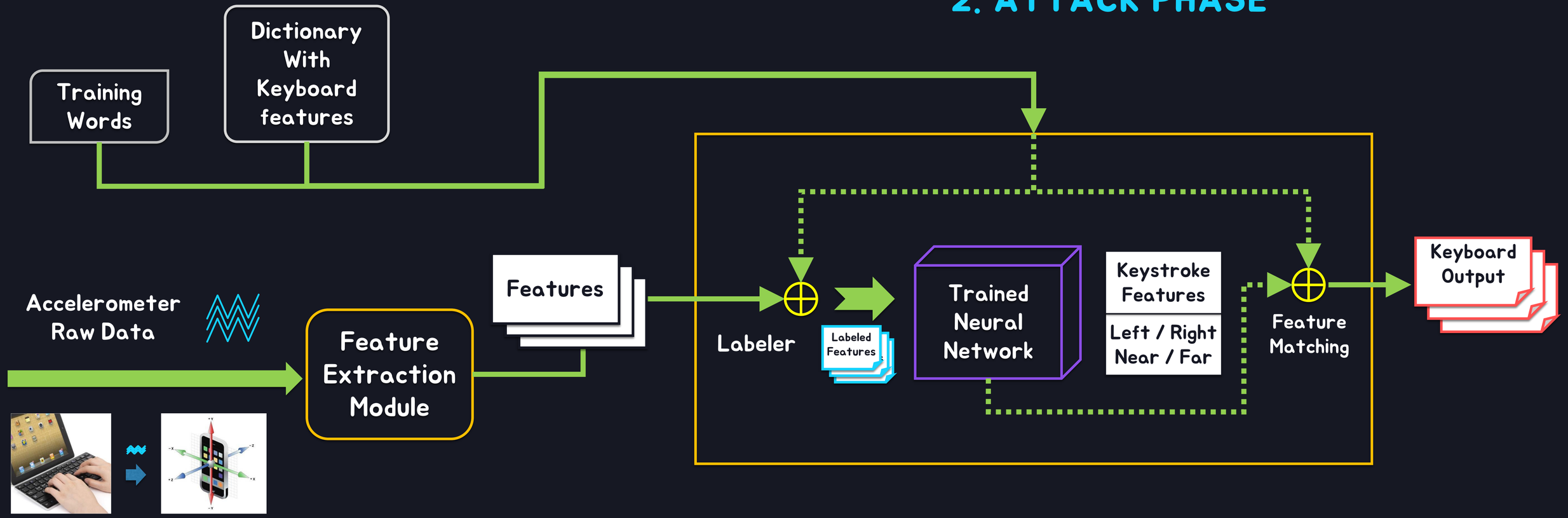
SmartPhone + Side Channel Attack +
Machine Learning = Stealing KeyStroke





ML Hacking Case #3

1. LEARNING PHASE
2. ATTACK PHASE



ML Hacking Case #3

1-5 Choice Correct = 80.00%
L/R Accuracy =78.58%
N/F Accuracy = 61.09%

L : LEFT R : RIGHT
N : NEAR F : FAR

Typed Text: The Illinois Supreme Court has ruled that Rahm Emanuel is eligible to run for mayor of Chicago and ordered him to stay on the ballot

실제 입력 내용

Recovered Text: *** Illinois Supreme about among might night Court *** ruled part wait Rahm Emanuel ** chicagos eligible **

*** *** names ** Chicago *** printed members grinned ordered *** ** look ** *** ballot

실험 결과

What should we do?

Intelligent Learning Data (Adversarial Training)

- 조작된 데이터를 Training Dataset에 포함시켜서 해당 공격에 대한 저항성을 가지게 함
- Regularization, SVM, 등등 ... 다양한 기법 응용

User Recognition

- AI 솔루션 != 만능 (AI 솔루션을 이용하더라도 잠재적 위협은 존재한다는 것을 인지)
- 모든 것을 솔루션에 위임하는 것 보다 사람의 검증이 있는 프로세스의 한 단계로 이용

THANK YOU