Welcome
2017 Faculty Summit Attendees

Faculty Summit 2017
microsoftfacultysummit.com

Microsoft Research
Microsoft.com/research

Facebook
@microsoftresearch

Twitter
@MSFTResearch
#FacSumm
#EdgeofAI

Microsoft Research
Faculty Summit **2017**

Microsoft Research
Faculty
Summit
2017

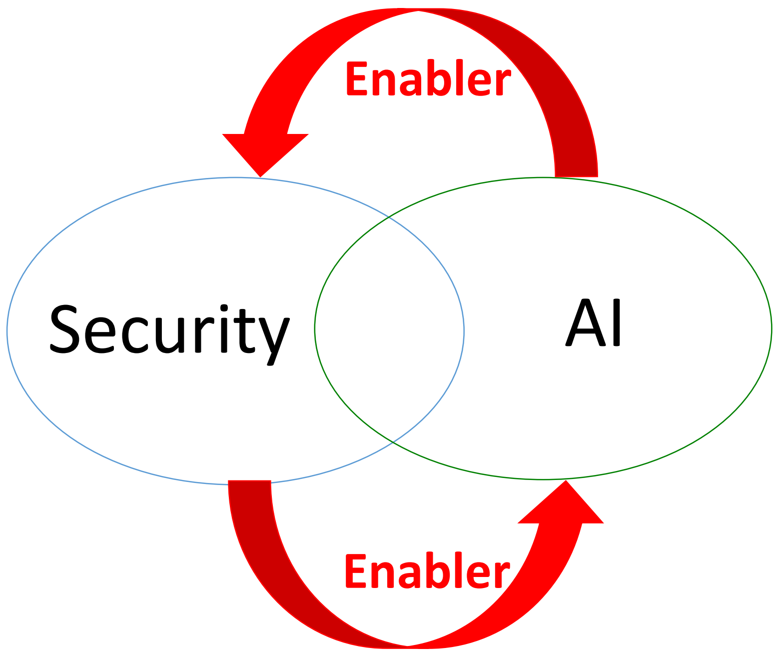
AI and Security

Questions for today

1) What can AI learn from security?
2) What can security learn from AI?
3) What does security look like after AI happens?

Dawn Song

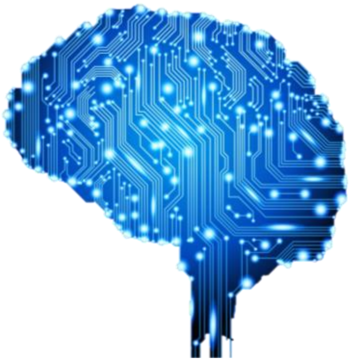
AI and Security



- Security enables better AI
  - Integrity: produces intended/correct results (adversarial machine learning)
  - Confidentiality/Privacy: does not leak users' sensitive data (secure, privacy-preserving machine learning)
  - Preventing misuse of AI
- AI enables security applications

AI and Security: AI in the presence of attacker

- Important to consider the presence of attacker
  - History has shown attacker always follows footsteps of new technology development (or sometimes even leads it)
  - The stake is even higher with AI
    - As AI controls more and more systems, attacker will have higher & higher incentives
    - As AI becomes more and more capable, the consequence of misuse by attacker will become more and more severe



AI and Security: AI in the presence of attacker

- Attack AI
  - Cause learning system to not produce intended/correct results
  - Cause learning system to produce targeted outcome designed by attacker
  - Learn sensitive information about individuals
  - Need security in learning systems

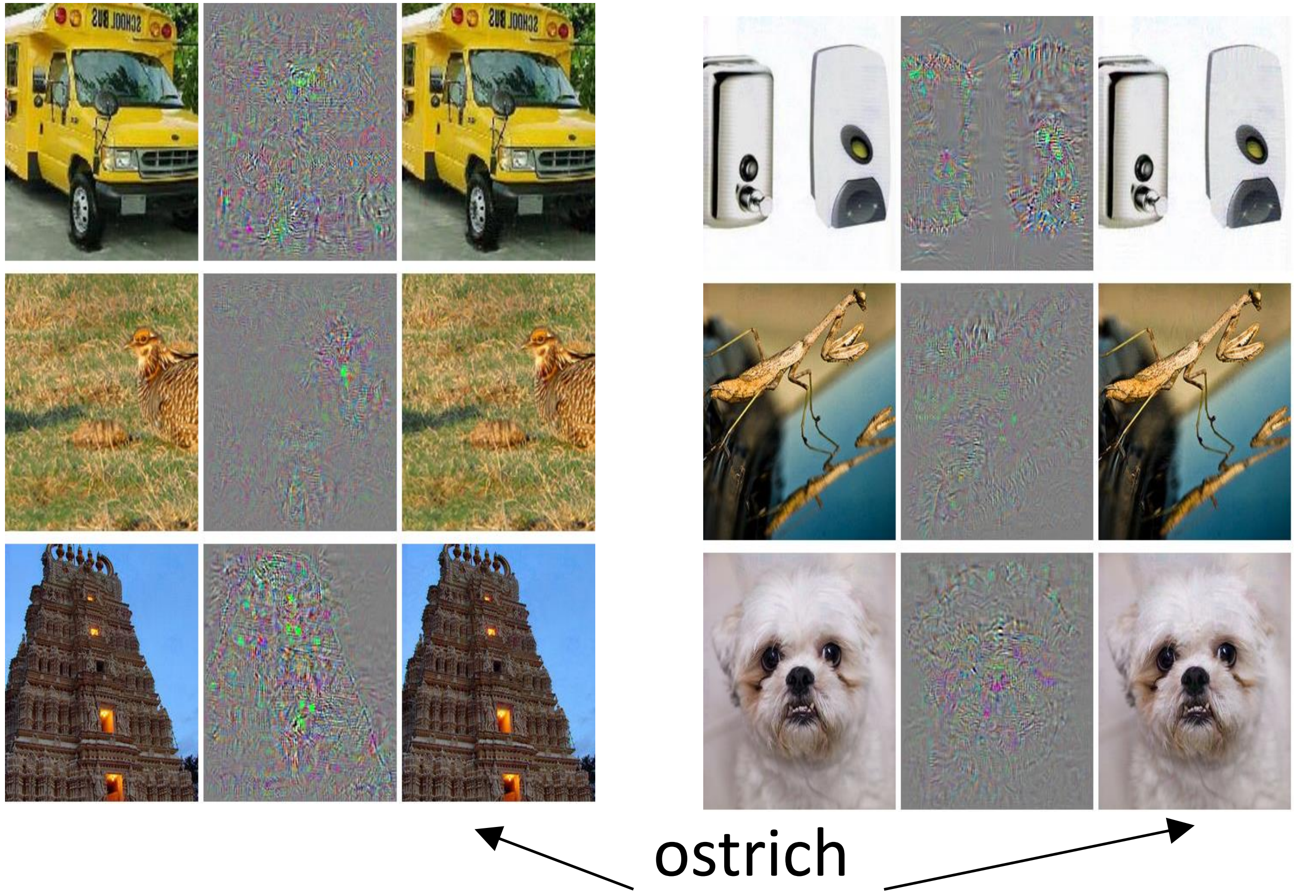
- Misuse AI
  - Misuse AI to attack other systems
    - Find vulnerabilities in other systems
    - Target attacks
    - Devise attacks
  - Need security in other systems

AI and Security: AI in the presence of attacker

- Attack AI:
  - Cause learning system to not produce intended/correct results
  - Cause learning system to produce targeted outcome designed by attacker
  - Learn sensitive information about individuals
  - Need security in learning systems

- Misuse AI
  - Misuse AI to attack other systems
    - Find vulnerabilities in other systems
    - Target attacks
    - Devise attacks
  - Need security in other systems

Deep Learning Systems Are Easily Fooled



$$\frac{\partial \text{output}}{\partial \text{pixels}}$$

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. Intriguing properties of neural networks. ICLR 2014.

Adversarial Examples Prevalent in Deep Learning Systems

- Most existing work on adversarial examples:
  - Image classification task
  - Target model is known

- Our investigation on adversarial examples:

Generative Models

Deep Reinforcement Learning

Image Captioning/ Image-to-code

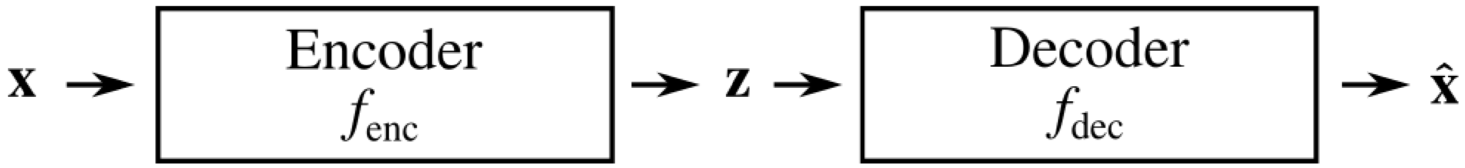
Other tasks and model classes

Blackbox Attacks

Weaker Threat Models
(Target model is unknown)

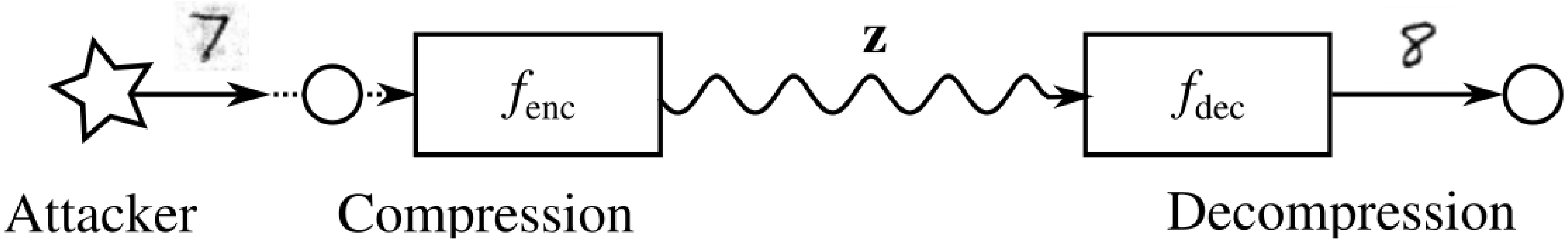
Generative models

- VAE-like models (VAE, VAE-GAN) use an intermediate latent representation
- An **encoder**: maps a high-dimensional input into lower-dimensional latent representation **z**.
- A **decoder:** maps the latent representation back to a high-dimensional reconstruction.



Adversarial Examples in Generative Models

- An example attack scenario:
  - Generative model used as a compression scheme



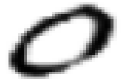
- Attacker’s goal: for the decompressor to reconstruct a different image from the one that the compressor sees.

Adversarial Examples for VAE-GAN in MNIST

Original images

Reconstruction of original images

Target Image



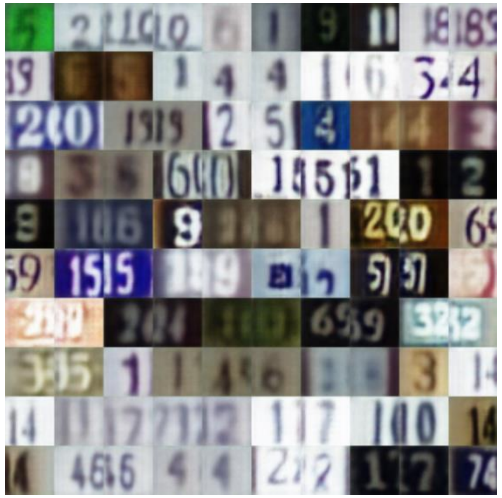
Adversarial examples

Reconstruction of adversarial examples

Jernej Kos, Ian Fischer, Dawn Song: Adversarial Examples for Generative Models

Adversarial Examples for VAE-GAN in SVHN

Original images



Reconstruction of original images

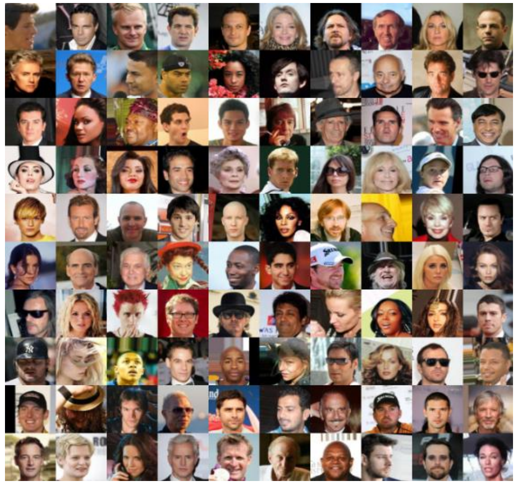
Target Image

Adversarial examples

Reconstruction of adversarial examples

Jernej Kos, Ian Fischer, Dawn Song: Adversarial Examples for Generative Models

Adversarial Examples for VAE-GAN in SVHN

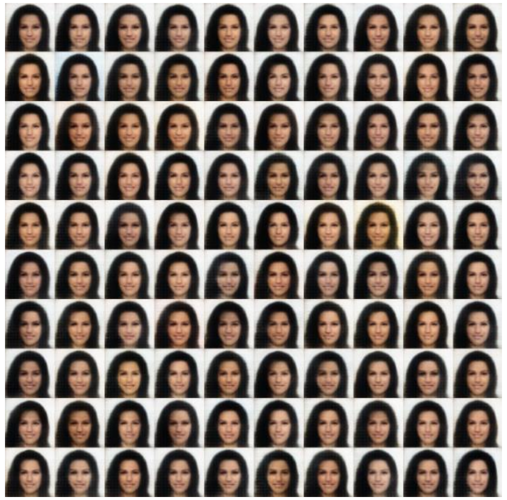


Original images

Reconstruction of original images

Target Image

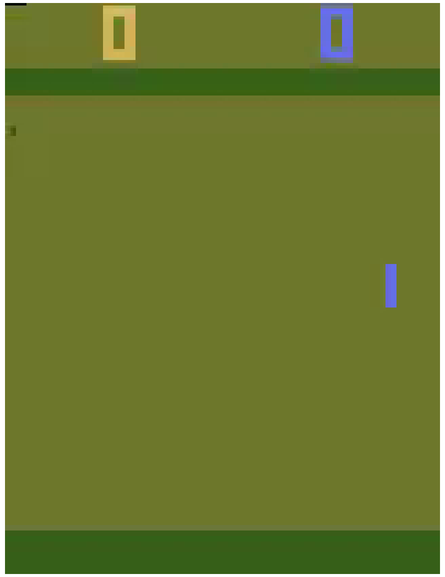
Adversarial examples



Reconstruction of adversarial examples

Jernej Kos, Ian Fischer, Dawn Song: Adversarial Examples for Generative Models

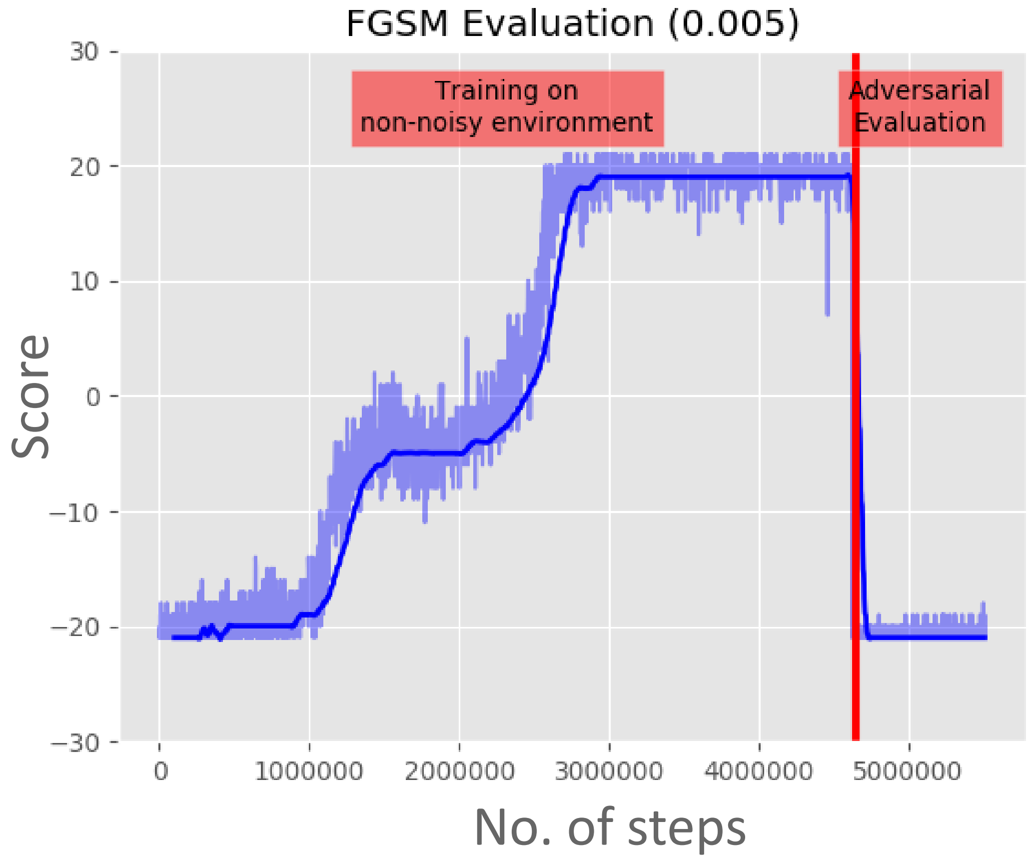
Deep Reinforcement Learning Agent (A3C) Playing Pong



Original Frames

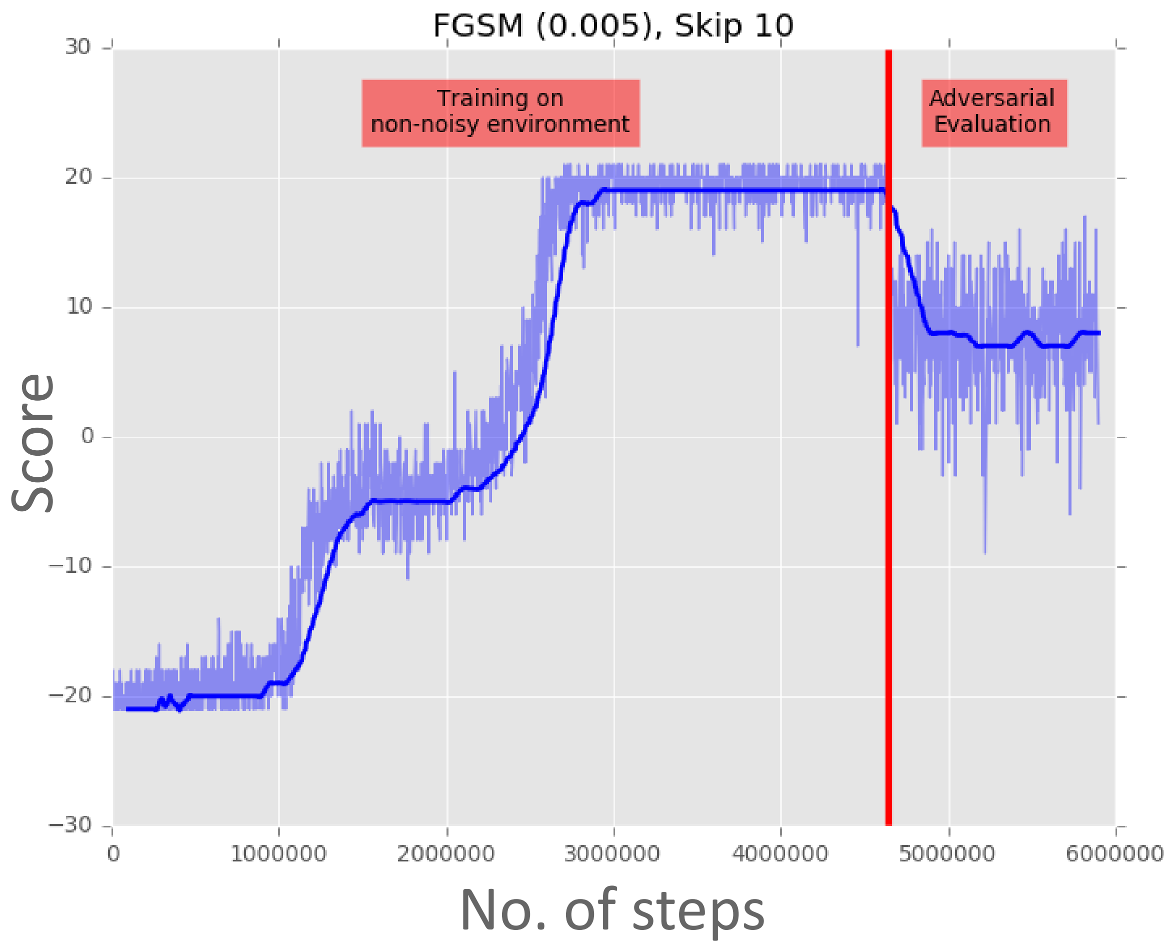
Jernej Kos and Dawn Song: Delving into adversarial attacks on deep policies [ICLR Workshop 2017].

Adversarial Examples on A3C Agent on Pong

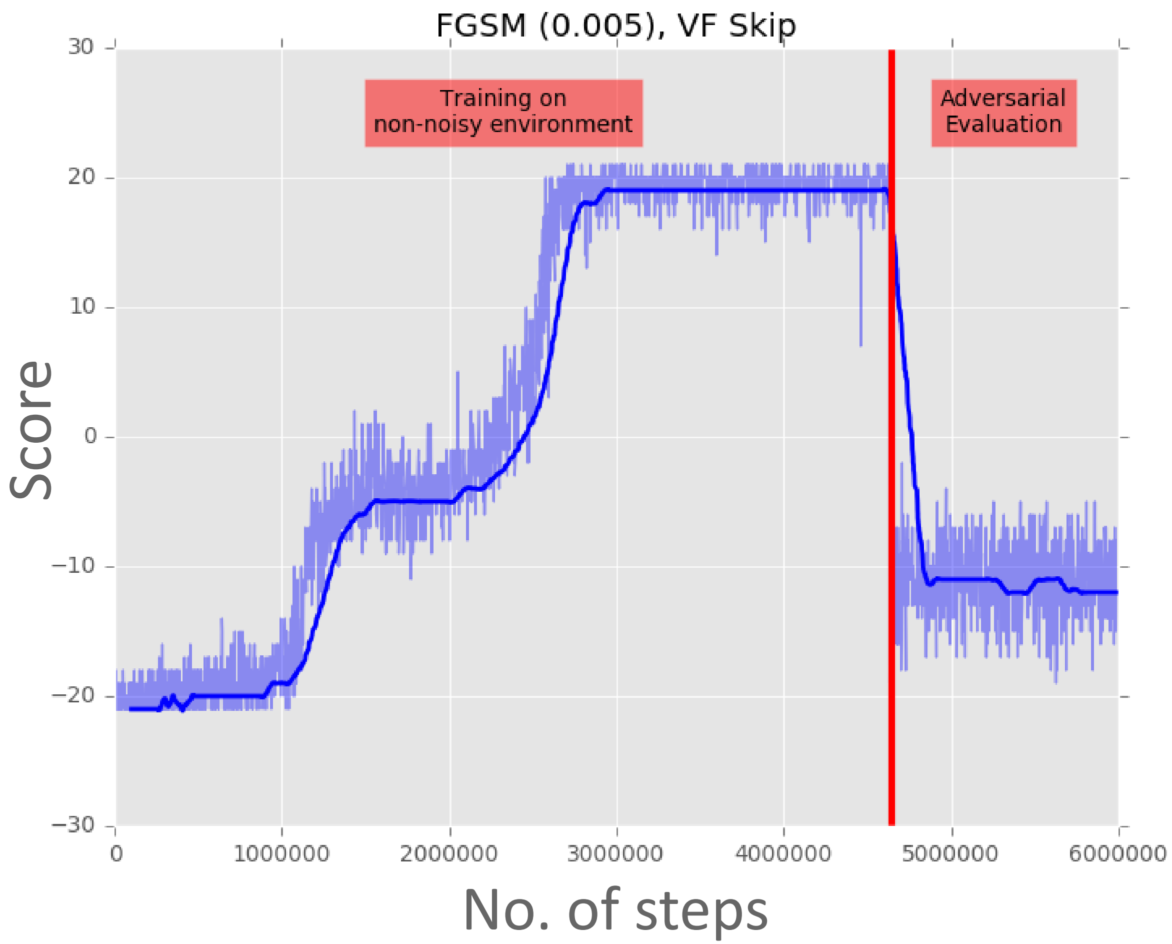


Jernej Kos and Dawn Song: Delving into adversarial attacks on deep policies [ICLR Workshop, 2017]

Attacks Guided by Value Function

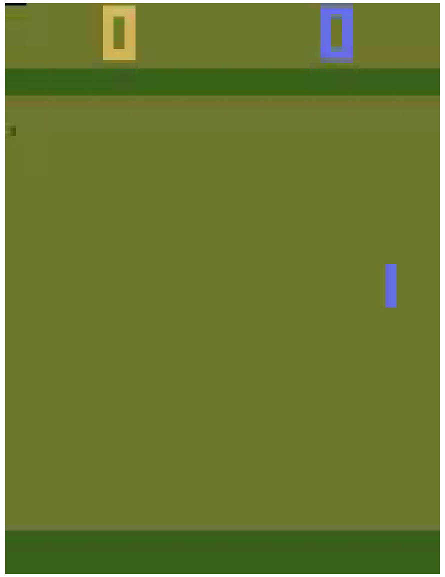


Blindly injecting adversarial perturbations every 10 frames.

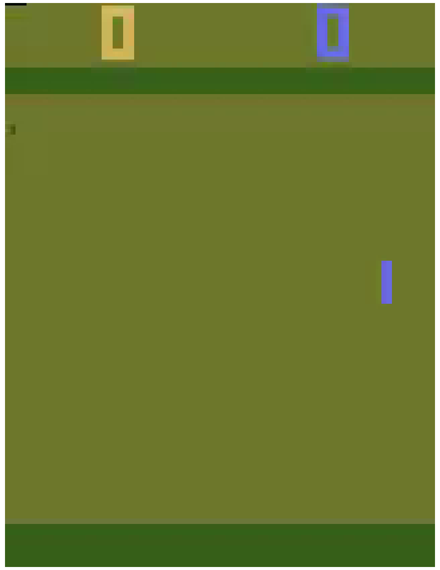


Injecting adversarial perturbations guided by the value function.

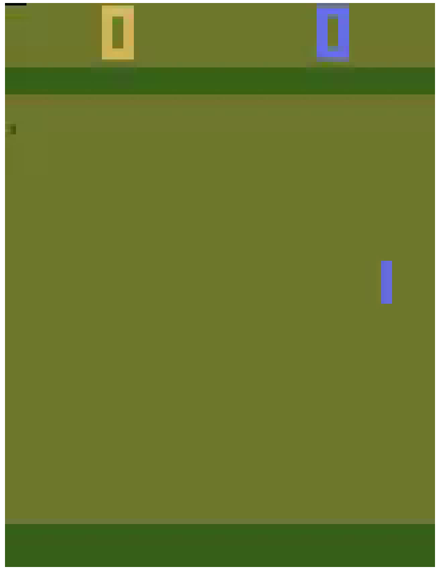
Agent in Action



Original Frames



With FGSM perturbations ($\epsilon = 0.005$) inject in every frame



With FGSM perturbations ($\epsilon = 0.005$) inject based on value function

Jernej Kos and Dawn Song: Delving into adversarial attacks on deep policies [ICLR Workshop 2017].

Adversarial Examples Prevalent in Deep Learning Systems

- Most existing work on adversarial examples:
  - Image classification task
  - Target model is known

- Our investigation on adversarial examples:

Generative Models

Deep Reinforcement Learning

Image Captioning/ Image-to-code

Other tasks and model classes

Blackbox Attacks

Weaker Threat Models
(Target model is unknown)

Adversarial Machine Learning

- Adversarial machine learning:
  - Learning in the presence of adversaries
- Inference time: adversarial example fools learning system
  - Evasion attacks
- Training time:
  - Attacker poisons training dataset (e.g., poison labels) to fool learning system to learn wrong model
    - Poisoning attacks: e.g., Microsoft's Tay twitter chatbot
  - Attacker selectively shows learner training data points (even with correct labels) to fool learning system to learn wrong model
- Adversarial machine learning particularly important for security critical systems

Security will be one of the biggest challenges in Deploying AI

Security of Learning Systems

- Software level
- Learning level
- Distributed level

Challenges for Security at Software Level

- No software vulnerabilities such as buffer overflows & access control issues
  - Attacker can take control over learning systems through exploiting software vulnerabilities

Challenges for Security at Software Level

- No software vulnerabilities (e.g., buffer overflows & access control issues)
- Existing software security/formal verification techniques apply

Reactive Defense

Proactive Defense:
Bug Finding

Proactive Defense:
Secure by Construction

Automatic worm detection
& signature/patch generation

Automatic malware
detection & analysis

Progression of my approach to software security over last 20 years

Challenges for Security at Learning Level

- Evaluate system under adversarial events, not just normal events

Regression Testing vs. Security Testing in Traditional Software System

	Regression Testing	**Security Testing**
Operation	Run program on **normal** inputs	Run program on **abnormal/adversarial** inputs
Goal	Prevent **normal** users from encountering errors	Prevent **attackers** from finding **exploitable** errors

Regression Testing vs. Security Testing in Learning System

	Regression Testing	**Security Testing**
Training	Train on noisy training data: Estimate resiliency against noisy training inputs	Train on poisoned training data: Estimate resiliency against poisoned training inputs
Testing	Test on **normal** inputs: Estimate generalization error	Test on **abnormal/adversarial** inputs: Estimate resiliency against adversarial inputs

Challenges for Security at Learning Level

- Evaluate system under adversarial events, not just normal events
  - Regression testing vs. security testing
- Reason about complex, non-symbolic programs

Decades of Work on Reasoning about Symbolic Programs

- Symbolic programs:
  - E.g., OS, File system, Compiler, web application, mobile application
  - Semantics defined by logic
  - Decades of techniques & tools developed for logic/symbolic reasoning
    - Theorem provers, SMT solvers
    - Abstract interpretation

Era of Formally Verified Systems

Verified: Micro-kernel, OS, File system, Compiler, Security protocols, Distributed systems

seL4 — Security. Performance. Proof.

IronClad/IronFleet

FSCQ

CertiKOS

miTLS/Everest

EasyCrypt

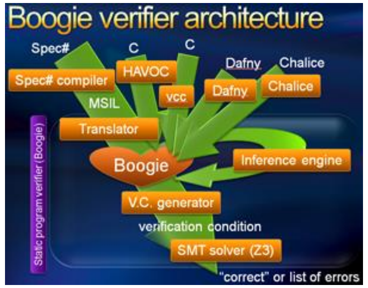
CompCert

Powerful Formal Verification Tools + Dedicated Teams

Coq

Why3

Z3



No Sufficient Tools to Reason about Non-Symbolic Programs

- Symbolic programs:
  - Semantics defined by logic
  - Decades of techniques & tools developed for logic/symbolic reasoning
    - Theorem provers, SMT solvers
    - Abstract interpretation

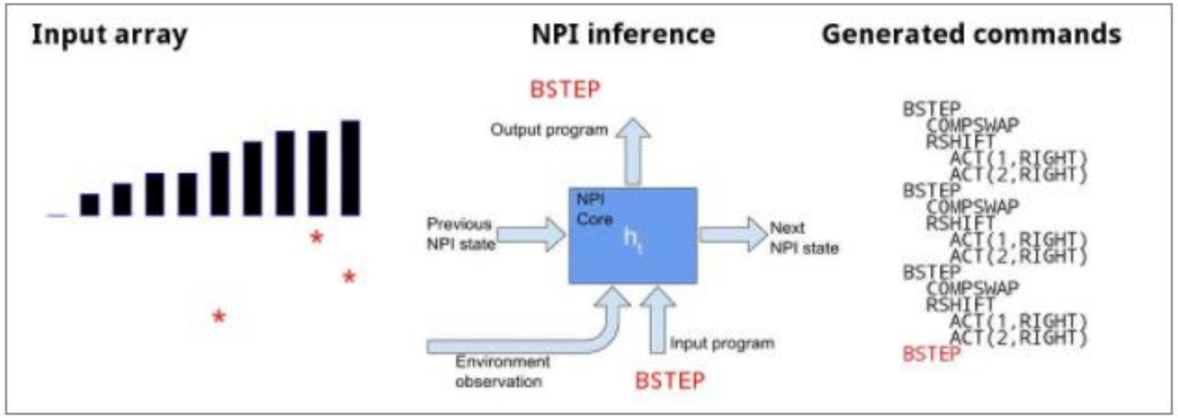
- Non-symbolic programs:
  - No precisely specified properties & goals
  - No good understanding of how learning system works
  - Traditional symbolic reasoning techniques do not apply

Challenges for Security at Learning Level

- Evaluate system under adversarial events, not just normal events
  - Regression testing vs. security testing
- Reason about complex, non-symbolic programs
- Design new architectures & approaches with stronger generalization & security guarantees

Limitation of Existing Neural Architectures

- **Example learning system**: Neural architectures learning programs
  - Neural Turing Machine, Neural GPU, Neural Random Access Machine, Differentiable Neural Computer
  - Neural Programmer Interpreter [Reed-Freitas, ICLR-2016, Best Paper Award]
  - Learn neural programs for addition, sorting, etc.



- **Problem**:
  - Neural architectures that learn programs currently do not generalize well (e.g., to problems of longer input length)
  - No provable guarantees about the generalization of the learned programs

Our Approach: Making Neural Programming Architectures Generalize via Recursion

- **Our Approach:**
    - Introduce notion of recursion to neural programs: ***Recursive neural programs***
    - Using recursion, a problem is reduced to *sub-problems*
        - Base cases and reduction rules
    - Learning recursive neural programs

Non-Recursive

```
ADD
  ADD1
    WRITE OUT 1
    CARRY
      PTR CARRY LEFT
      WRITE CARRY 1
      PTR CARRY RIGHT
  LSHIFT
    PTR INP1 LEFT
    PTR INP2 LEFT
    PTR CARRY LEFT
    PTR OUT LEFT
  ADD1
    ...
```

Recursive

```
ADD
  ADD1
    WRITE OUT 1
    CARRY
      PTR CARRY LEFT
      WRITE CARRY 1
      PTR CARRY RIGHT
  LSHIFT
    PTR INP1 LEFT
    PTR INP2 LEFT
    PTR CARRY LEFT
    PTR OUT LEFT
  ADD
    ...
```

Jonathon Cai, Richard Shin, Dawn Song: Making Neural Programming Architectures Generalize via Recursion [ICLR 2017, **Best Paper Award**]

Our Approach: Making Neural Programming Architectures Generalize via Recursion

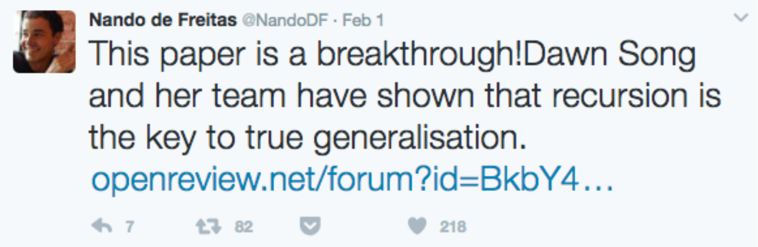
- **Proof of Generalization**:
  - Recursion enables provable guarantees about neural programs
  - Prove perfect generalization of a learned recursive program via a verification procedure
    - Explicitly testing on all possible base cases and reduction rules (Verification set)

$$\forall i \in V(S), M(i) \Downarrow P(i)$$

- Learn & generalize faster as well
  - Trained on same data, non-recursive programs do not generalize well

Accuracy on Random Inputs for Quicksort

Length of Array	Non-Recursive	Recursive
3	100%	100%
5	100%	100%
7	100%	100%
11	73.3%	100%
15	60%	100%
20	30%	100%
22	20%	100%
25	3.33%	100%
30	3.33%	100%
70	0%	100%



Jonathon Cai, Richard Shin, Dawn Song: Making Neural Programming Architectures Generalize via Recursion [ICLR 2017, **Best Paper Award**]

Lessons

- Program architecture impacts provability:
  - Similar in program verification for symbolic programs
  - Well-designed programs with good architectures are easier to prove properties of
  - Arbitrary programs (bad code) are difficult to prove properties of
- Caution for end-to-end monolithic neural networks
  - Harder to train
  - Harder to generalize
  - Harder to interpret
- Recursive, modular neural architectures are easier to reason, prove, generalize
- Explore new architectures and approaches enabling strong generalization & security properties for broader tasks
  - For complex perception tasks, what should we do?
- Can we have provable guarantee of generalization & security properties for general learning systems?

Challenges for Security at Learning Level

- Evaluate system under adversarial events, not just normal events
- Reason about complex, non-symbolic programs
- Design new architectures & approaches with stronger generalization & security guarantees
- Reason about how to compose components

Compositional Reasoning

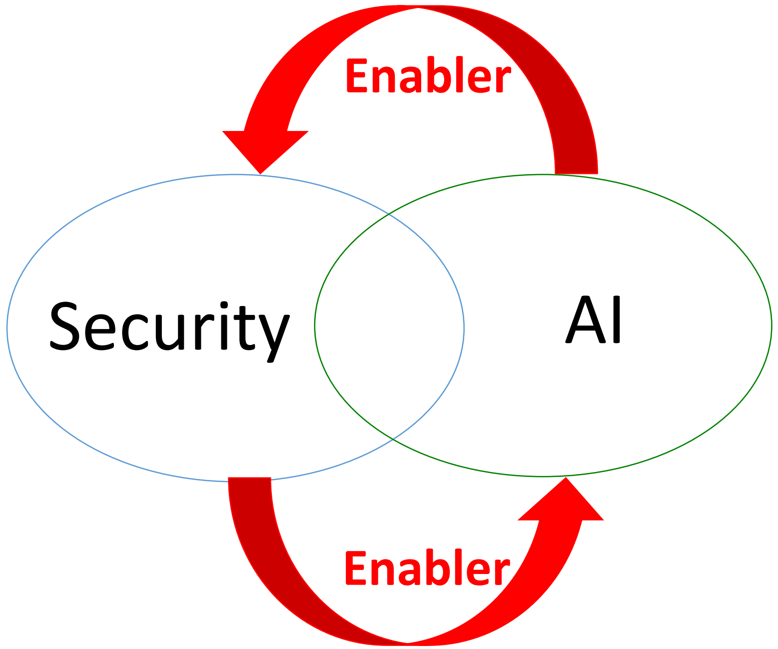
- Building large, complex systems require compositional reasoning
  - Each component provides abstraction
    - E.g., pre/post conditions
  - Hierarchical, compositional reasoning proves properties of whole system

- How to do abstraction, compositional reasoning for non-symbolic programs?

Security of Learning Systems

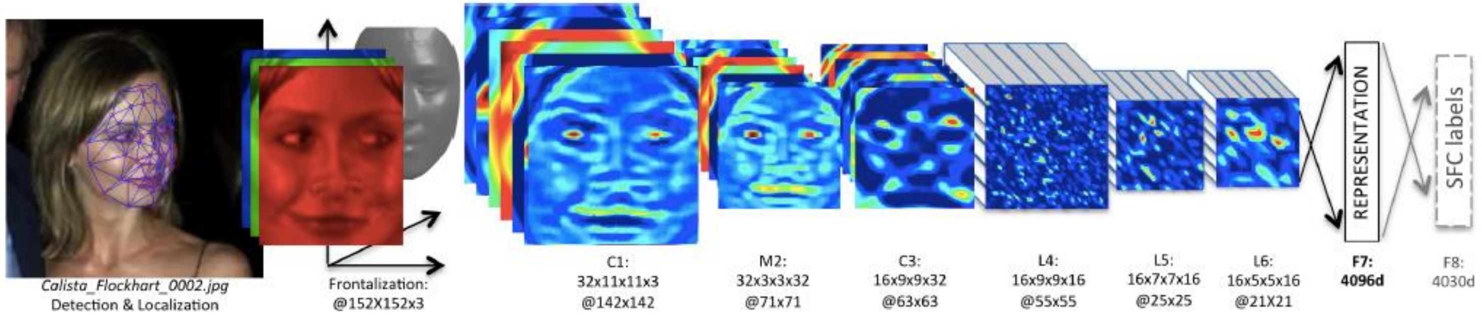
- Software level
- Learning level
  - Evaluate system under adversarial events, not just normal events
  - Reason about complex, non-symbolic programs
  - Design new architectures & approaches with stronger generalization & security guarantees
  - Reason about how to compose components
- Distributed level
  - Each agent makes local decisions; how to make good local decisions achieve good global decision?

AI and Security



- Security enables better AI
  - Integrity: produces intended/correct results (adversarial machine learning)
  - Confidentiality/Privacy: does not leak users' sensitive data (secure, privacy-preserving machine learning)
  - Preventing misuse of AI
- AI enables security applications

Deep Learning Improving Security Capabilities



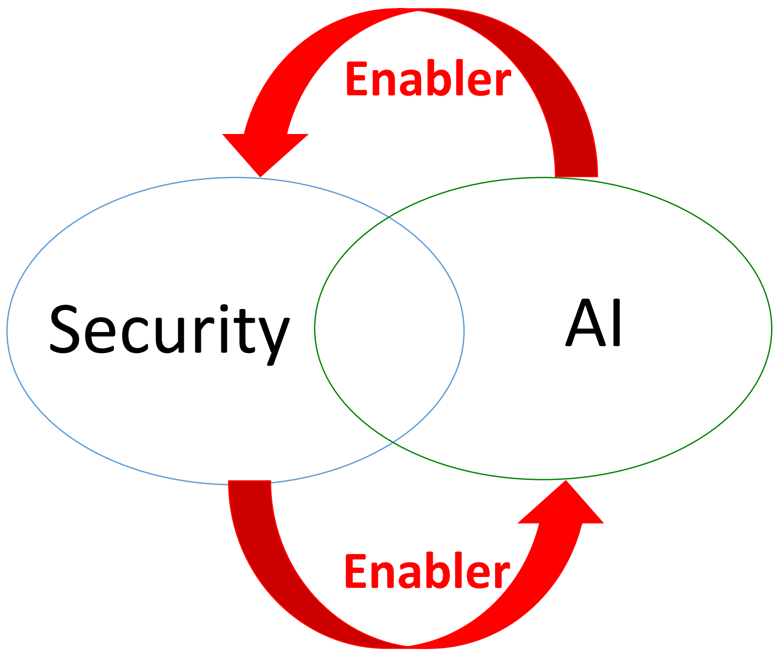
DeepFace

When/where is machine learning applicable in security applications?

Learning is Most Needed When No Precise Formal Property Specification

- Example:
  - Spam filtering
  - Fraud detection
  - Account compromise
  - Bots vs. human
  - In contrast to memory-safety exploits detection & defense, etc.
- Property specification depends on fuzzy concepts & world model
- Symbolic reasoning does not apply
- Need learning-based approach

AI and Security



- Security enables better AI
  - Integrity: produces intended/correct results (adversarial machine learning)
  - Confidentiality/Privacy: does not leak users' sensitive data (secure, privacy-preserving machine learning)
  - Preventing misuse of AI
- AI enables security applications

AI and Security: AI in the presence of attacker

- Attack AI
  - Cause learning system to not produce intended/correct results
  - Cause learning system to produce targeted outcome designed by attacker
  - Need security in learning systems

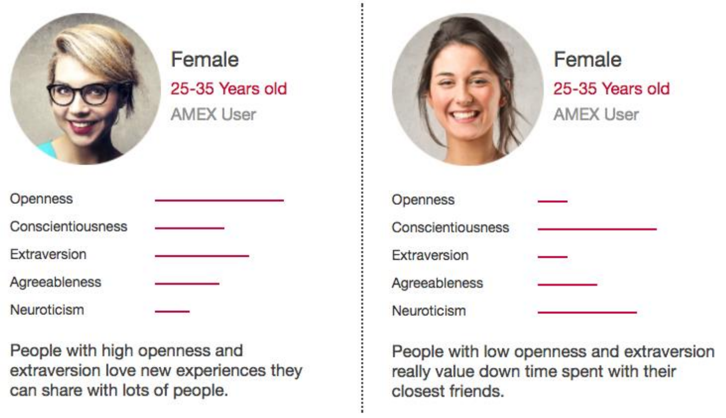
- Misuse AI
  - Misuse AI to attack other systems
    - Find vulnerabilities in other systems
    - Target attacks
    - Devise attacks
  - Need security in other systems

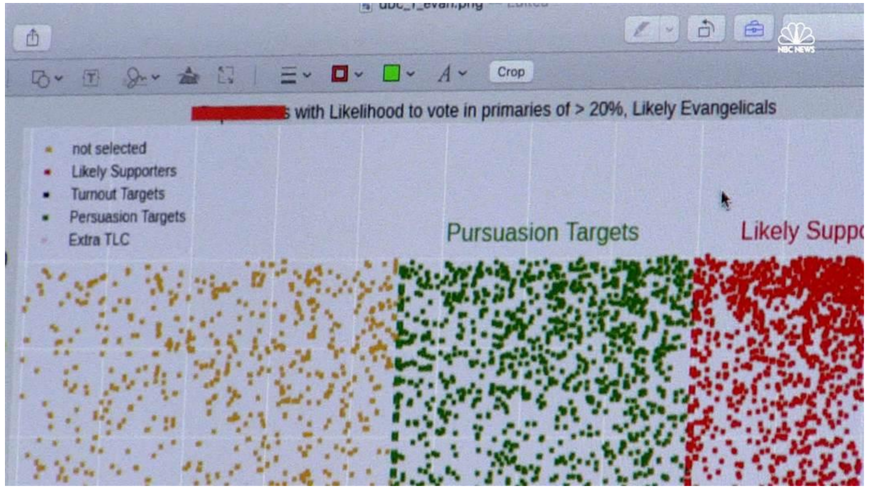
Misused AI can make attacks more effective

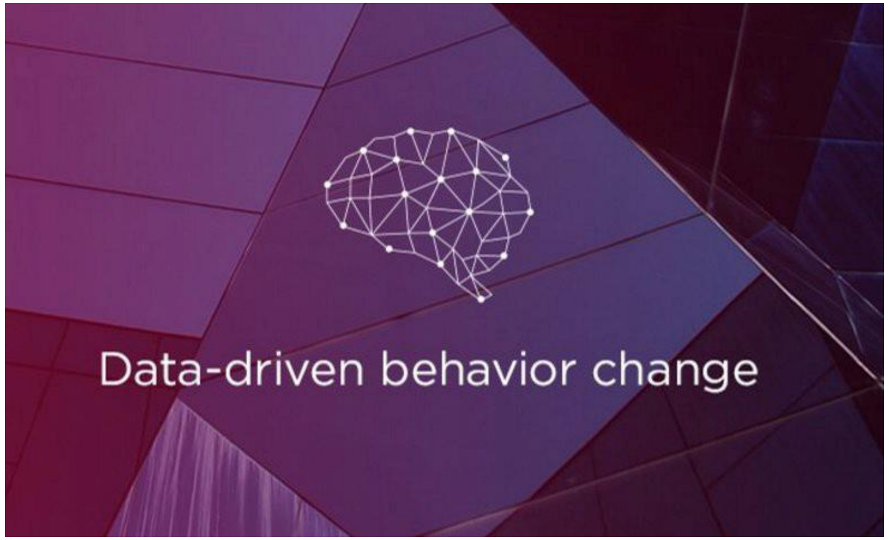
Deep Learning Empowered
Bug Finding

Deep Learning Empowered
Phishing Attacks

Misused AI for large-scale, automated, targeted manipulation







Consumer-grade BCI Devices

- Price: ≈ 300 USD

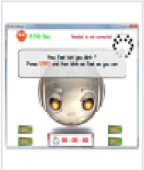
HEADSET & ACCESSORIES

DEVELOPER & RESEARCH PACKAGES

APP STORE

Exercise Equipment for Your Mind

Experts agree that the human brain should be exercised like other body elements. Use the MindWave with specially designed neuroscience meditation, mental fitness and game applications on your home PC or Mac.



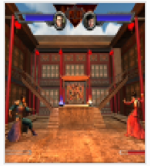
BLINKCHALLENGE

Uses a Emobot interface and it can catch your blink immediately. Try to beat your longest stare! Or how fast can you blink? You just wear the headset and try this game

Rate this product:

$4.95

BUY NOW



ARENA

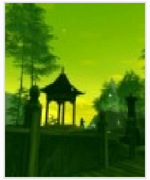
This is a game that requires you to use the power of your mind against your opponent. To play the game, you must first train your mind to shoot fireballs using the Emotiv PUSH command.

This game supports single and dual player modes. For dual player mode (DUEL) each player will

Rate this product:

$14.95

BUY NOW



SPIRIT MOUNTAIN DEMO GAME

Experience the fantasy of having supernatural powers and controlling the world with your mind. Your journey will take you through a mythical landscape of forests, temples and an environment that adjusts itself based on how you feel.

Rate this product:

FREE

FREE

DOWN LOAD

MASTER MIND

Master Mind allows users to play their favorite PC games with the power of their mind. Existing PC games such as World of Warcraft™ and Call of Duty™ can now be played with the power of your mind.

$99.00

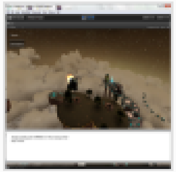
BUY NOW

MIND MOUSE

Mind Mouse is a revolutionary thought-controlled software application which allows the user to navigate the computer, click and double click to open programs, compose email and send with the power of their mind. *** "NON 'AA

$99.00

BUY NOW



EMOTIV EPOC UNITY3D™ DEVELOPER SUPPORT PACK

This package contains a full Unity3D™ Wrapper for the Emotiv EPOC EmoEngine API and a working demonstration game project and assets.

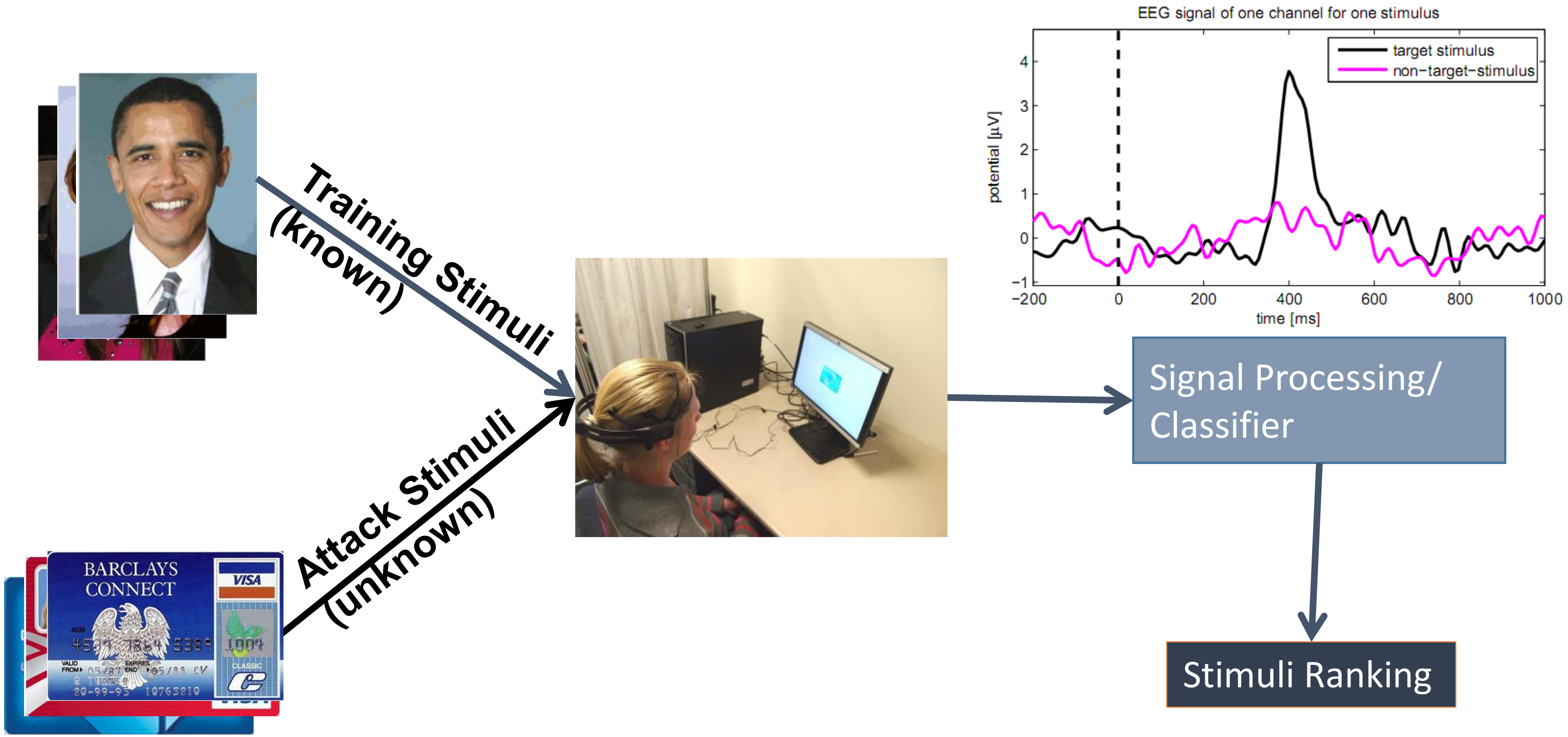
$79.95

BUY NOW

What if an EEG gaming app is malicious?

Secretly reading your mind?

BCI as Side-Channel to the Brain



On the Feasibility of Side−Channel Attacks with Brain−Computer Interfaces [USENIX Security’12]

Attack Stimuli

(a) ATM

(b) Debit Card

Information tested & learned:

- First digit of PIN
- Do you know this person?
- Do you have an account at this bank?
- What month were you born in?
- Where do you live?

On the Feasibility of Side−Channel Attacks with Brain−Computer Interfaces [USENIX Security’12]

The Dual

The More Powerful Consumer-grade BCI devices are

The More Powerful AI technology is

The More Powerful the attacks are

With great power comes great responsibility

Marvel.com

Lessons from Medical Device Security

- First medical device security analysis in public literature:
  - *The case for Software Security Evaluations of Medical Devices [HRMPFS, HealthSec'11]*

- FDA issues guidance recommendation on medical device security [2016]

Security will be one of the biggest challenges in Deploying AI

Important to consider security for AI from early on

Secure AI is important and necessary for future advancement of AI

Secure AI is an interdisciplinary, community effort

Future of AI and Security

How to better understand what security means for AI, learning systems?

How to detect when a learning system has been fooled/compromised?

How to build more resilient systems with stronger guarantees?

How to mitigate misuse of AI?

What should be the right policy to ensure secure AI?

dawnsong@cs.berkeley.edu

Let’s tackle the big challenges together!

Taesoo Kim

About Myself

Taesoo Kim (taesoo@gatech.edu)

- 14-   : Assistant Professor at Georgia Tech
- 11-14: Ph.D. from MIT in CS

Research interests:

Operating Systems, Systems Security, Distributed Systems, Programming Languages, Architecture

https://taesoo.kim

Clarification: Security and AI

- Security → Software or Computer Security
  - In particular, attacker's perspectives
  - Excluding the security issues that involved human (e.g., fraud, phishing …)

- AI → ML or Deep Learning
  - In particular, training-based, stochastic approaches
  - It works well in practice, but too complex to understand why? or how?

Three Key Points

- Part 1. What AI can learn from Security?

  → Thinking like an adversary

- Part 2. What Security can learn from AI?

  → Measuring the progress of research

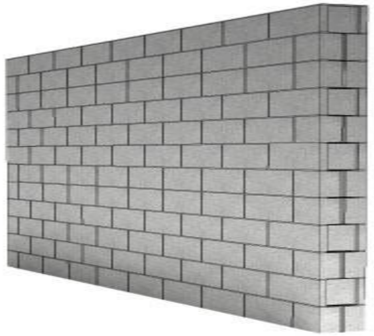
- Part 3. Security after AI?

  → New Era for Advanced Persistent Threats (APT)

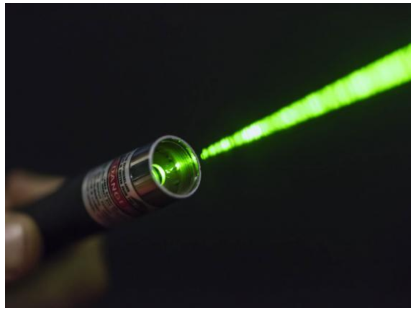
Part 1. What AI can learn from Security? Thinking Like an Adversary

How to hijack this self-driving car?

Putting wall?



Attacking sensors?



Put STOP signs?

Part 1. What AI can learn from Security? Thinking Like an Adversary

Laying a trap for self-driving cars

Posted Mar 17, 2017 by Devin Coldewey

Adversary? Meeting with the Best Hacker!

The Register®
Biting the hand that feeds IT

ENTER SOFTWARE SECURITY TRANSFORMATION DEVOPS BUSINESS PERSONAL TECI

Security 19

***Reg* meets 'Lokihardt', quite possibly the world's best hacker**

Korean chap finds flaws in moments, scores $100k apiece for fun

By Darren Pauli 11 Nov 2016 at 07:22 SHARE ▾

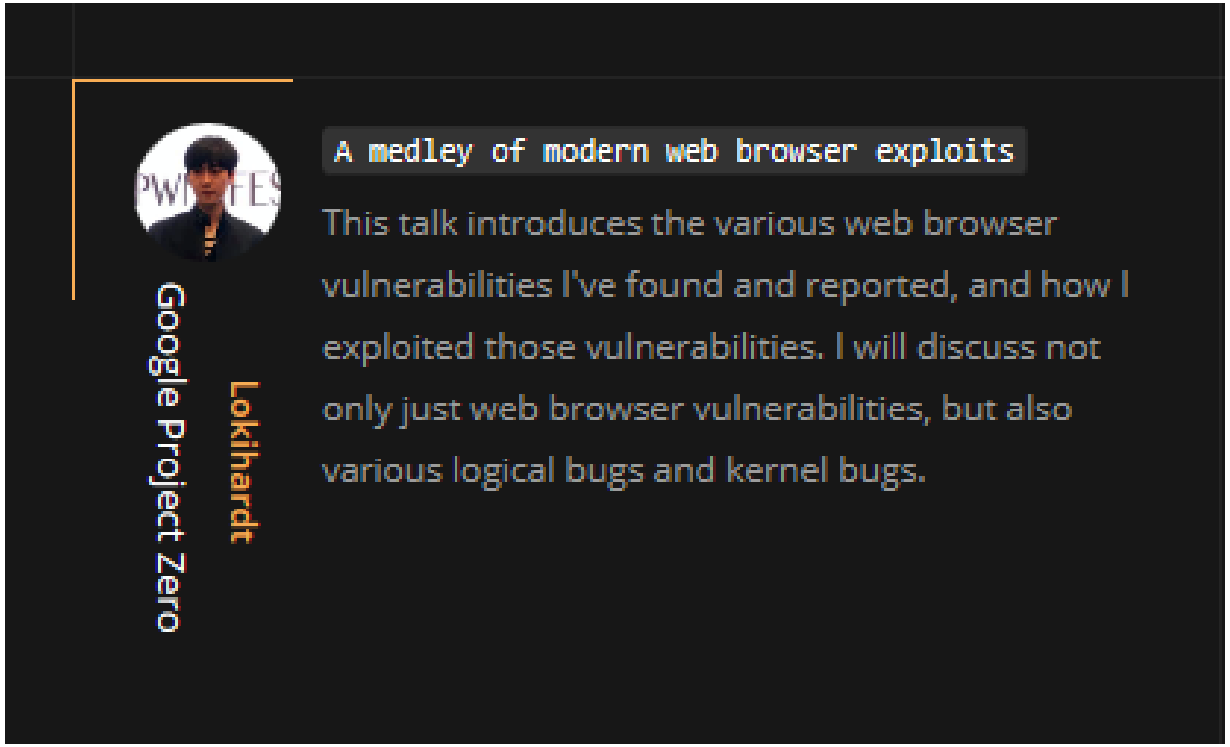
Full-chain exploitation on all major browsers and platforms!

$225,000 in Pwn2Own'15
$300,000 in PwnFest'16
…

Now in Google's Project Zero Team

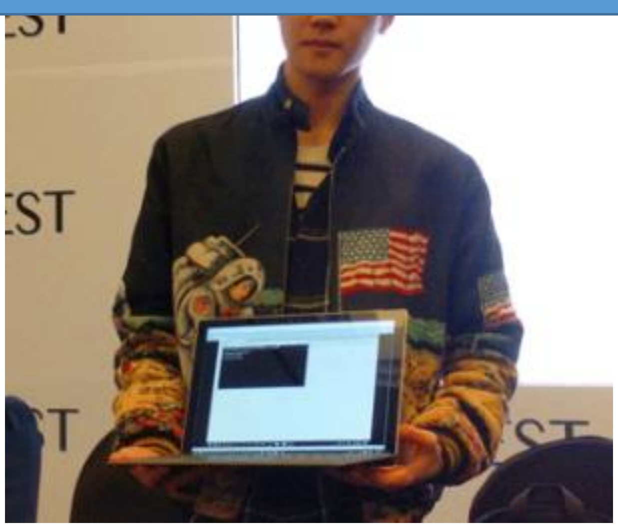
First Public Talk @Zer0Con'17

Conference for Exploit Developers & Bug Hunters



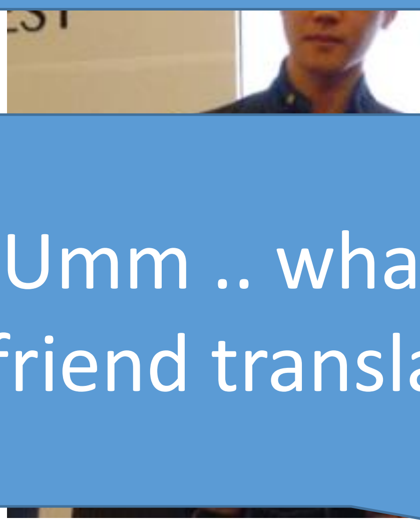
Lots of (even) Hackers are Curious ..

Could you explain how you found bugs in Pwn2Own'16?



Lots of (even) Hackers are Curious ..

Could you explain how you found bugs in Pwn2Own'16?



Umm .. what?
(his friend translated ..)

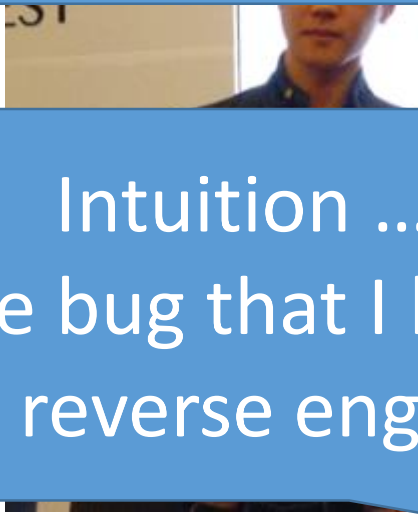
Lots of (even) Hackers are Curious ..

Could you explain how you found bugs in Pwn2Own’16?

“Intuition …”

Lots of (even) Hackers are Curious ..

Could you explain how you found bugs in Pwn2Own'16?



Intuition …
except one bug that I had to open IDA for reverse engineering.

Approaches to Security vs. ML

- Security:
  (Translating) Intuition → Methodologies

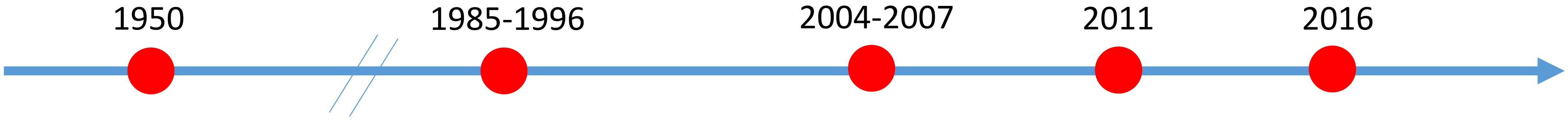
vs.

- ML:
  (Inferring) Training data → Parameters

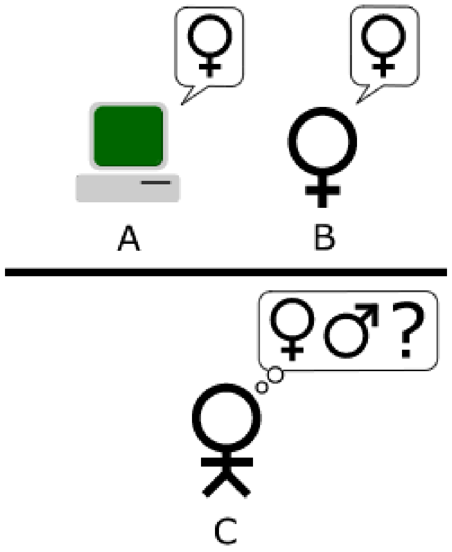
Take-away Messages from Security

- Attackers target a single, weakest component
- Rethinking of your assumption (aka, threat model)
- Increasing #features → larger attack surface
- Focusing on *directly* translating intuition to models
- Making the design iteration comprehensive (ie., explainable)

Part 2. What Security can learn from AI?
Measuring the Progress of Research



Turing Test



Chess

Deep Blue

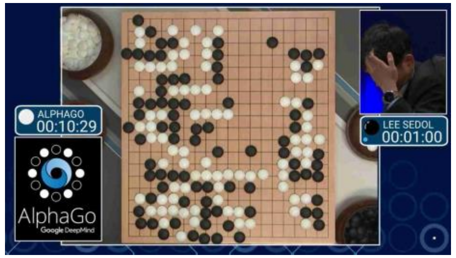
Driving

DARPA Grand Challenge

Jeopardy

Watson

Go!

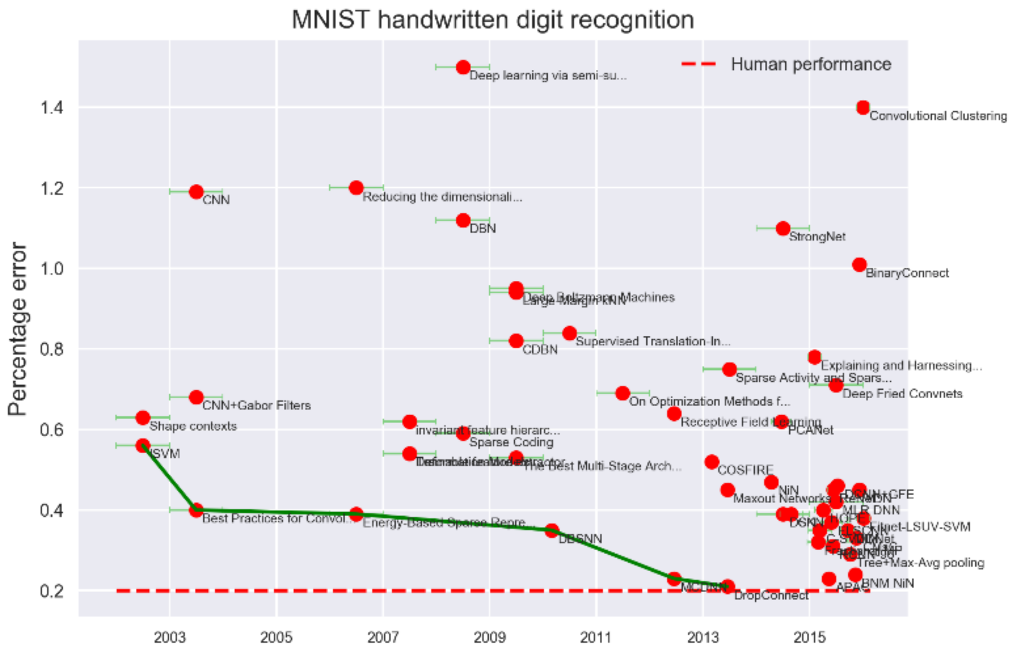


AlphaGo

Electronic Frontier Foundation (EFF) announces:

https://www.eff.org/ai/metrics

(e.g., handwritten digit recognition)



What happens to Security (and Privacy)?

- Perhaps, too subject?
- What do you even mean by measuring “security”?

- In terms of exploit/defenses:

CTF games
(human vs. human)

DARPA Cyber Grand Challenge
(computer vs. computer)

Take-away Messages from AI

- AI fields drive research as various landmark competitions
- Public resources for quantifying the progress (e.g., data sets)
- Perhaps, people tend to “hide” security-related data
- Too subjective, but we might be able to tackle subfields of security?
- So we can objectively measure pros/cons of security mechanisms

Part 3. Security After AI:
New Era for Advanced Persistent Threats

It works, but I don't know why?

- AI takes off → "unknown" software everywhere!
- In particular, when Security relies on AI-based approaches

(APT = Advanced persistent threat, or targeted attack)


The Real Story of Stuxnet

How Kaspersky Lab tracked down the malware that stymied Iran's nuclear-fuel enrichment program

By DAVID KUSHNER
Posted 26 Feb 2013 | 14:00 GMT

Computer cables snake across the floor. Cryptic flowcharts are scrawled across various whiteboards adorning the walls. A life-size Batman doll stands in the hall. This office might seem no different than any other geeky workplace, but in fact it's the front line of a war—a cyberwar, where most battles play out not in remote jungles or deserts but in

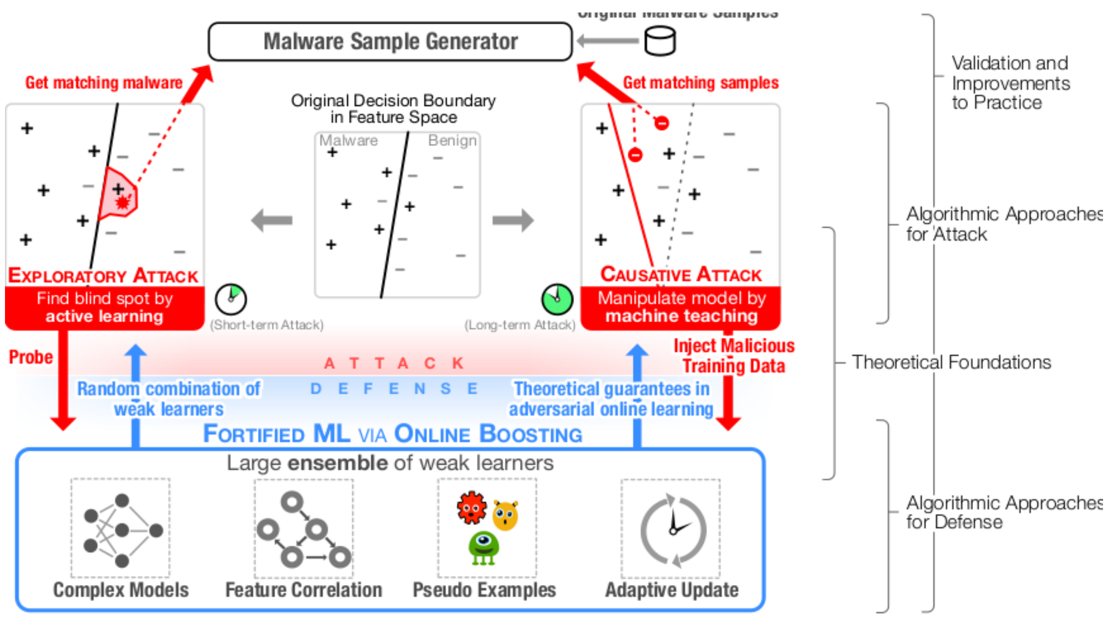
Take-away Messages (once AI takes off)

- More attack surface for attackers: impl, algorithm, data, etc.

- What if attackers understand more deeply than you?
- What if attackers can influence your data set?
- What if we don't even observe attacks (i.e., accountability)?

On-going Efforts at Georgia Tech

- Intel Science and Technology Center (ISTC) for Adversary-Resilient Security Analytics (MLsploit)
- NSF/SaTC: CORE: Medium: Understanding and Fortifying Machine Learning Based Security Analytics
    - Security: Wenke Lee, Taesoo Kim
    - ML: Polo Chau, Le Song



Mike Walker

“What can AI learn from security”?

1996

```
.oO Phrack 49 Oo.

Volume Seven, Issue Forty-Nine

File 14 of 16

BugTraq, r00t, and Underground.Org
bring you

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Smashing The Stack For Fun And Profit
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

by Aleph One
aleph1@underground.org
```

2011

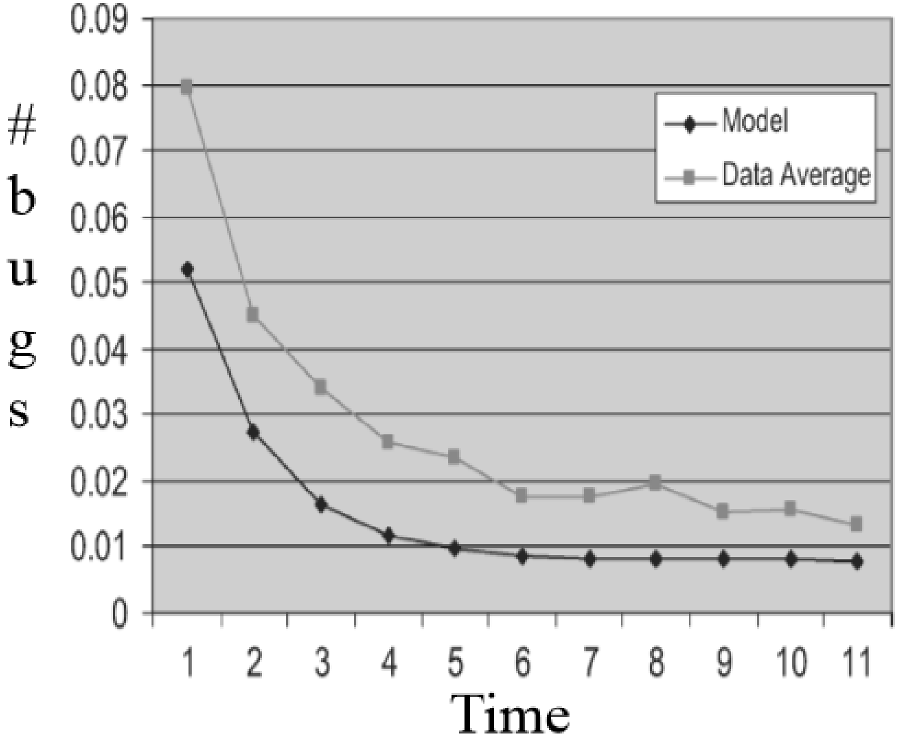
Memory Corruption (19)	
Defeated by DEP	14
Defeated by ASLR	17
Defeated by EMET	19

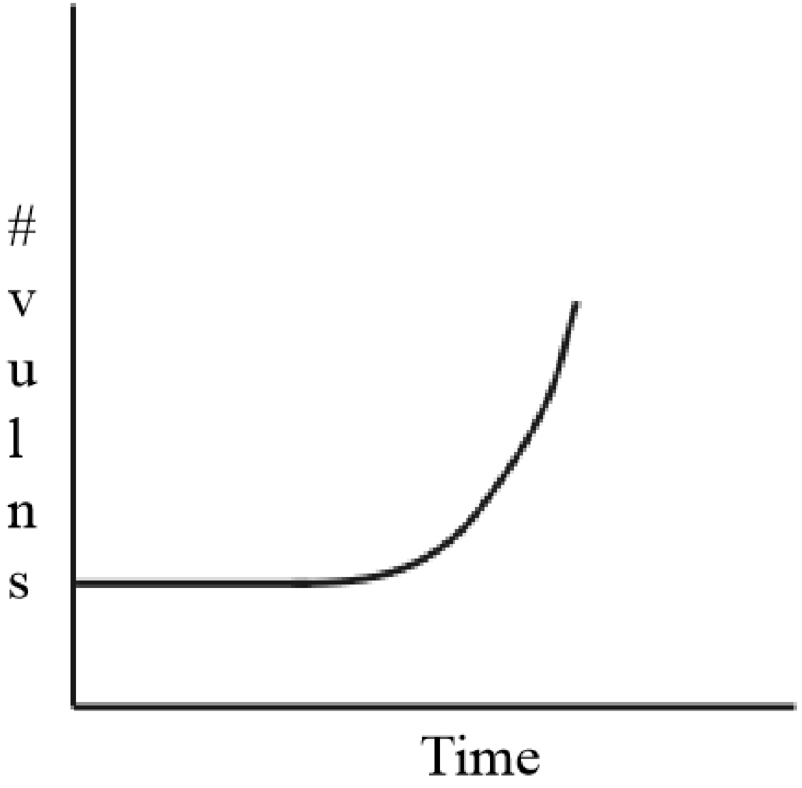
Logic Flaws (8)	
No Java in Internet Zone	4
No EXEs in PDFs	1
No Firefox or FoxIt Reader	2

$114B

Right: Dan Guido, Exploit Intelligence Project
Left: Aleph One, Phrack 49

The Honeymoon Effect



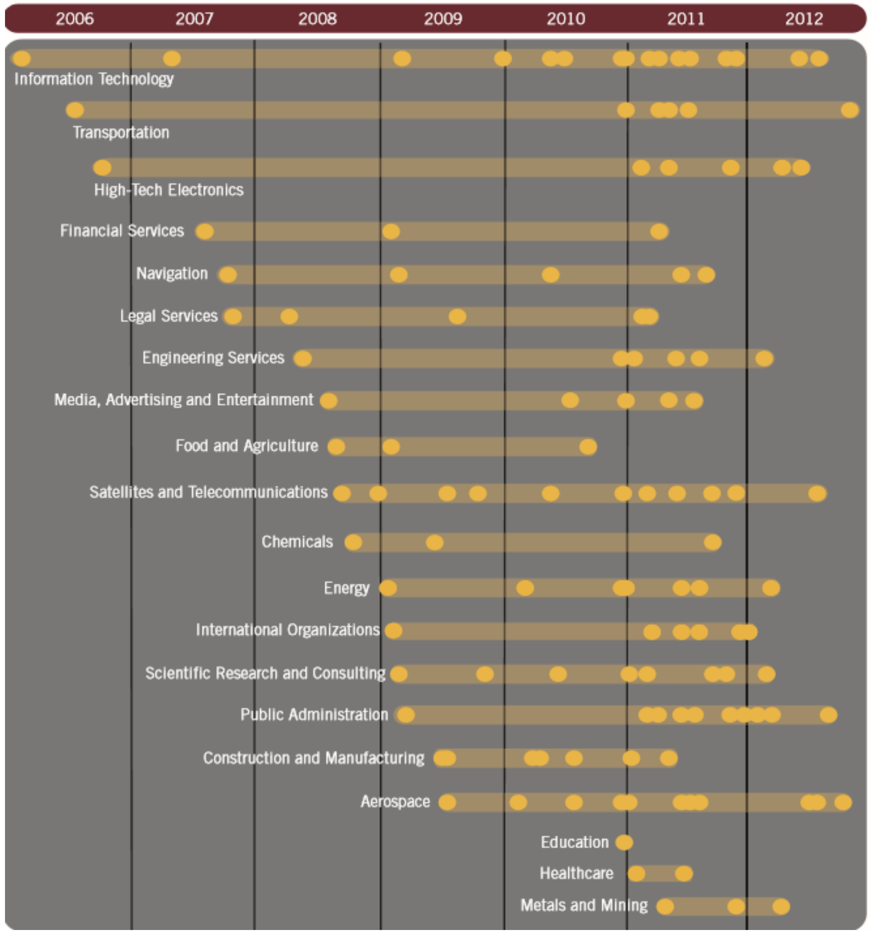


Bugs: Starts fast,
then ***slows down***

Vulnerabilities: Starts slow,
then ***speeds up!***

Frei, Clark, Blaze, Smith: Familiarity Breeds Contempt

No Reports of Attacks



Mandiant, APT1 Report

Golden Opportunities in AI Security

- Any software that serves as a gatekeeper to valuable IP, wealth, or life safety must consider the eventual arrival of an expert adversary
- Attack detection is not free; it requires active research & sensors
- No reports of attacks != no attacks
- Techniques to defeat security properties must be discovered & published in the open first (Fun) or be exploited (Profit)

“What can security learn from AI?”

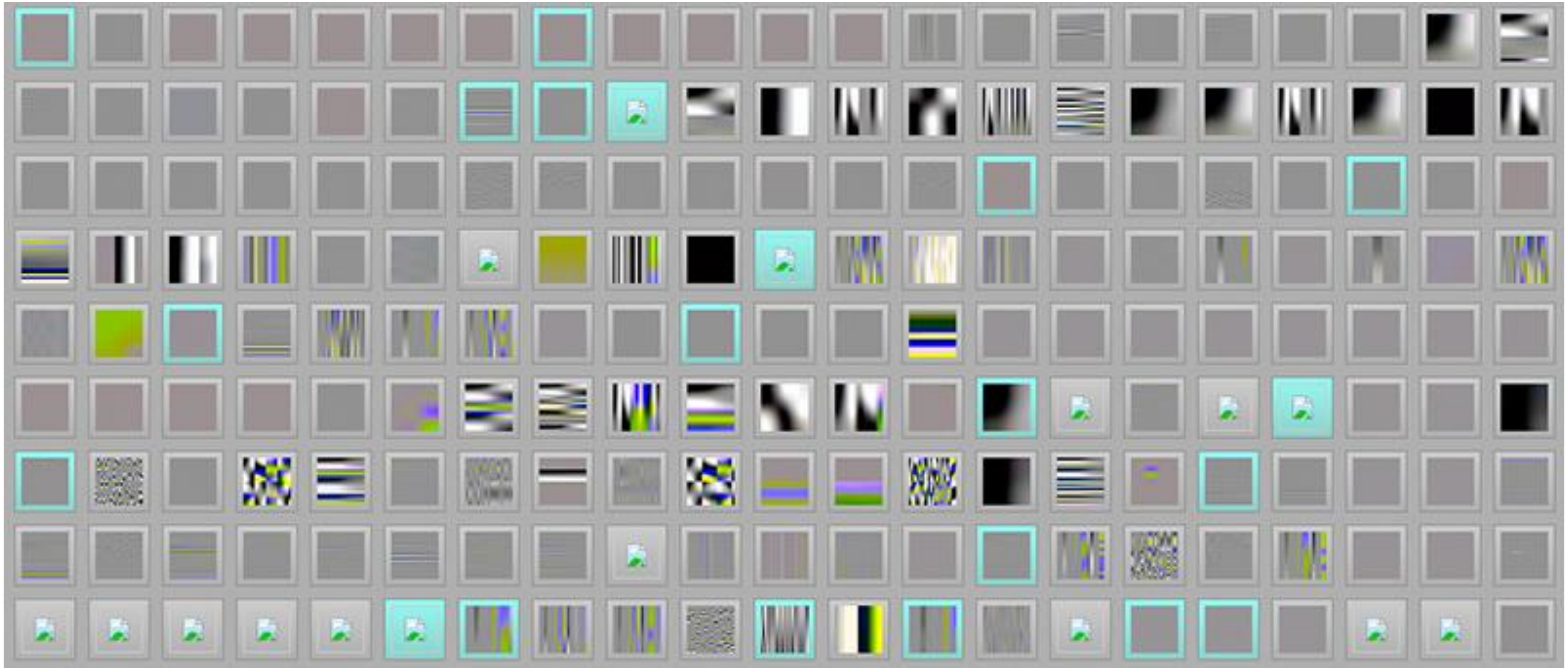
Machine Learning versus Sensors

<u>SandPrint:</u>

“we can use those inherent features to detect sandboxes using supervised machine learning techniques […] an attacker can reveal characteristics of publicly available sandboxes and use the gathered information to build a classifier that can perfectly distinguish between a user PC and an appliance”

http://christian-rossow.de/publications/sandprint-raid2016.pdf

AFL vs. djpeg



Michal Zalewski, Google, on AFL

“Security & AI”


CodeJitsu
GALACTICA
CodeJitsu
TEAM CSDS
JIMA
CYBER SECURITY DEVELOPMENT SOLUTIONS
SPONSORED BY
University of Idaho
SCORING
CYBER
RUBEUS
disekt
CYBER
SECURE
DARPA
CYBER
DARPA
CYBER
DARPA
DARPA
DARPA
CYBER
DARPA

Discussion

“What can AI learn from security”?

“What can security learn from AI?”

“What does security look like after AI ‘happens’ ? ”

Wrap-up and next steps

- What can AI learn from security?
- What can security learn from AI?
- What does security look like after AI happens?

New techniques, new problems to solve, new collaborations

Find someone to work with today!

Thank you

Microsoft