

SECURITY AND PRIVACY

Eunsuk Kang

Required reading: *Building Intelligent Systems: A Guide to Machine Learning Engineering*, G. Hulten (2018), Chapter 25: Adversaries and Abuse. *The Top 10 Risks of Machine Learning Security*, G. McGraw et al., IEEE Computer (2020).

LEARNING GOALS

- Explain key concerns in security (in general and with regard to ML models)
- Identify security requirements with threat modeling
- Analyze a system with regard to attacker goals, attack surface, attacker capabilities
- Describe common attacks against ML models, including poisoning and evasion attacks
- Understand design opportunities to address security threats at the system level
- Apply key design principles for secure system design

SECURITY: (VERY BRIEF) OVERVIEW

ELEMENTS OF SECURITY

ELEMENTS OF SECURITY

- Security requirements (also called "policies")
 - What does it mean for my system to be secure?

ELEMENTS OF SECURITY

- Security requirements (also called "policies")
 - What does it mean for my system to be secure?
- Threat model
 - What are the attacker's goals, capabilities, and incentives?

ELEMENTS OF SECURITY

- Security requirements (also called "policies")
 - What does it mean for my system to be secure?
- Threat model
 - What are the attacker's goals, capabilities, and incentives?
- Attack surface
 - Which parts of the system are exposed to the attacker?

ELEMENTS OF SECURITY

- Security requirements (also called "policies")
 - What does it mean for my system to be secure?
- Threat model
 - What are the attacker's goals, capabilities, and incentives?
- Attack surface
 - Which parts of the system are exposed to the attacker?
- Defense mechanisms (mitigations)
 - How do we prevent the attacker from compromising a security requirement?

SECURITY REQUIREMENTS



- What do we mean by "secure"?
- Common security requirements: "CIA triad" of information security
- **Confidentiality:** Sensitive data must be accessed by authorized users only
- **Integrity:** Sensitive data must be modifiable by authorized users only
- **Availability:** Critical services must be available when needed by clients

EXAMPLE: COLLEGE ADMISSION SYSTEM

FEATURE

Hacker helps applicants breach security at top business schools

Among the institutions affected were Harvard, Duke and Stanford

Using the screen name "brookbond," the hacker broke into the online application and decision system of ApplyYourself Inc. and posted a procedure students could use to access information about their applications before acceptance notices went out.

CONFIDENTIALITY, INTEGRITY, OR AVAILABILITY?

CONFIDENTIALITY, INTEGRITY, OR AVAILABILITY?

- Applications to the program can only be viewed by staff and faculty in the department.

CONFIDENTIALITY, INTEGRITY, OR AVAILABILITY?

- Applications to the program can only be viewed by staff and faculty in the department.
- The application site should be able to handle requests on the day of the application deadline.

CONFIDENTIALITY, INTEGRITY, OR AVAILABILITY?

- Applications to the program can only be viewed by staff and faculty in the department.
- The application site should be able to handle requests on the day of the application deadline.
- Application decisions are recorded only by the faculty and staff.

CONFIDENTIALITY, INTEGRITY, OR AVAILABILITY?

- Applications to the program can only be viewed by staff and faculty in the department.
- The application site should be able to handle requests on the day of the application deadline.
- Application decisions are recorded only by the faculty and staff.
- The acceptance notices can only be sent out by the program director.

OTHER SECURITY REQUIREMENTS

- Authentication: Users are who they say they are
- Non-repudiation: Certain changes/actions in the system can be traced to who was responsible for it
- Authorization: Only users with the right permissions can access a resource/perform an action

THREAT MODELING

WHY THREAT MODEL?



WHAT IS THREAT MODELING?

- Threat model: A profile of an attacker
 - **Goal:** What is the attacker trying to achieve?
 - **Capability:**
 - Knowledge: What does the attacker know?
 - Actions: What can the attacker do?
 - Resources: How much effort can it spend?
 - **Incentive:** Why does the attacker want to do this?



"If you know the enemy and know yourself, you need not fear the result of a hundred battles."
- Sun Tzu, *The Art of War*

ATTACKER GOAL

- What is the attacker trying to achieve?

ATTACKER GOAL

- What is the attacker trying to achieve?
 - Typically, undermine one or more security requirements

ATTACKER GOAL

- What is the attacker trying to achieve?
 - Typically, undermine one or more security requirements
- Example: College admission

ATTACKER GOAL

- What is the attacker trying to achieve?
 - Typically, undermine one or more security requirements
- Example: College admission
 - Access other applicants info without being authorized

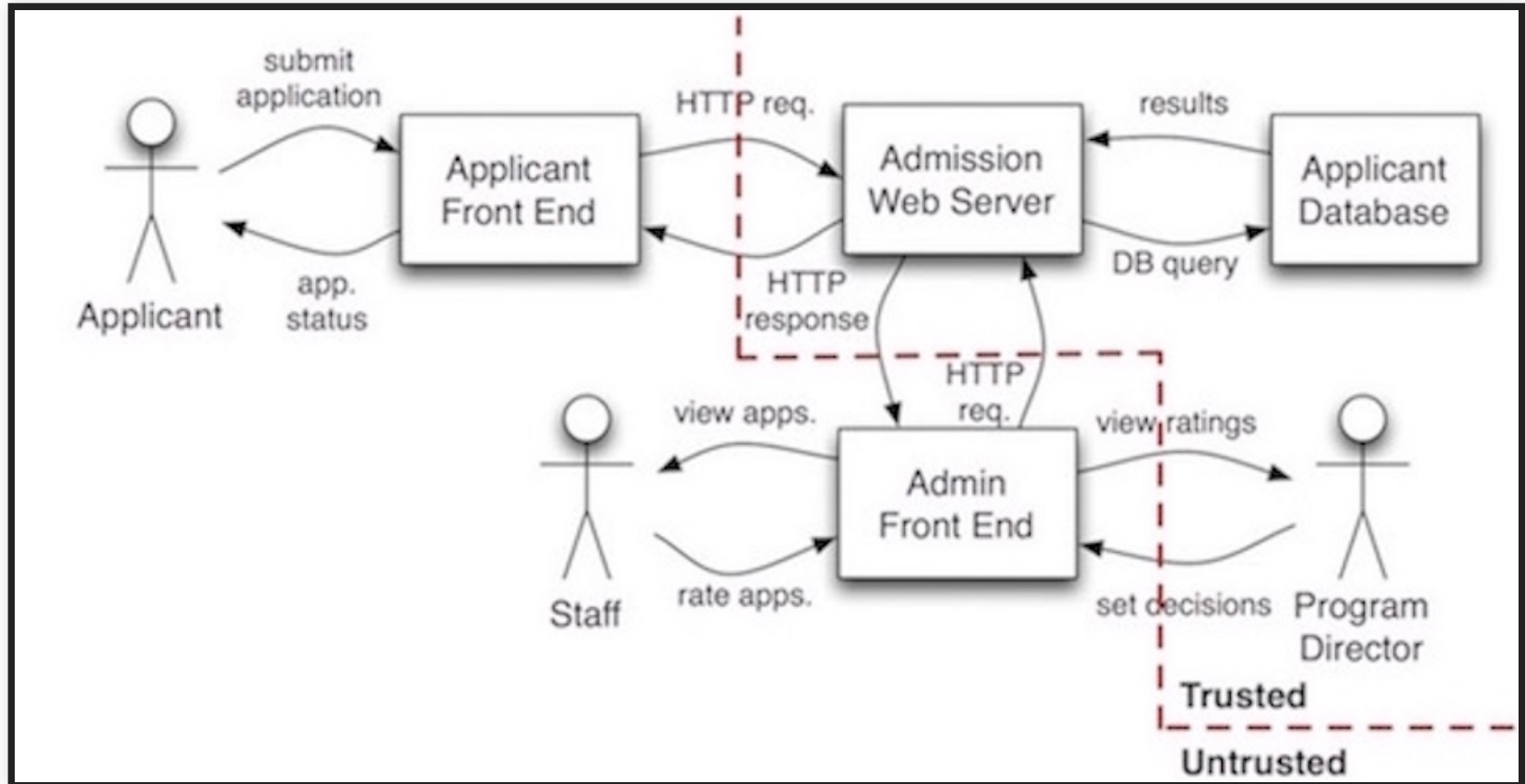
ATTACKER GOAL

- What is the attacker trying to achieve?
 - Typically, undermine one or more security requirements
- Example: College admission
 - Access other applicants info without being authorized
 - Modify application status to “accepted”

ATTACKER GOAL

- What is the attacker trying to achieve?
 - Typically, undermine one or more security requirements
- Example: College admission
 - Access other applicants info without being authorized
 - Modify application status to “accepted”
 - Cause website shutdown to sabotage other applicants

ATTACKER CAPABILITY



- What actions are available to the attacker (to achieve its goal)?
 - Depends on system boundary & interfaces exposed to external actors
 - Use an architecture diagram to identify attack surface & actions

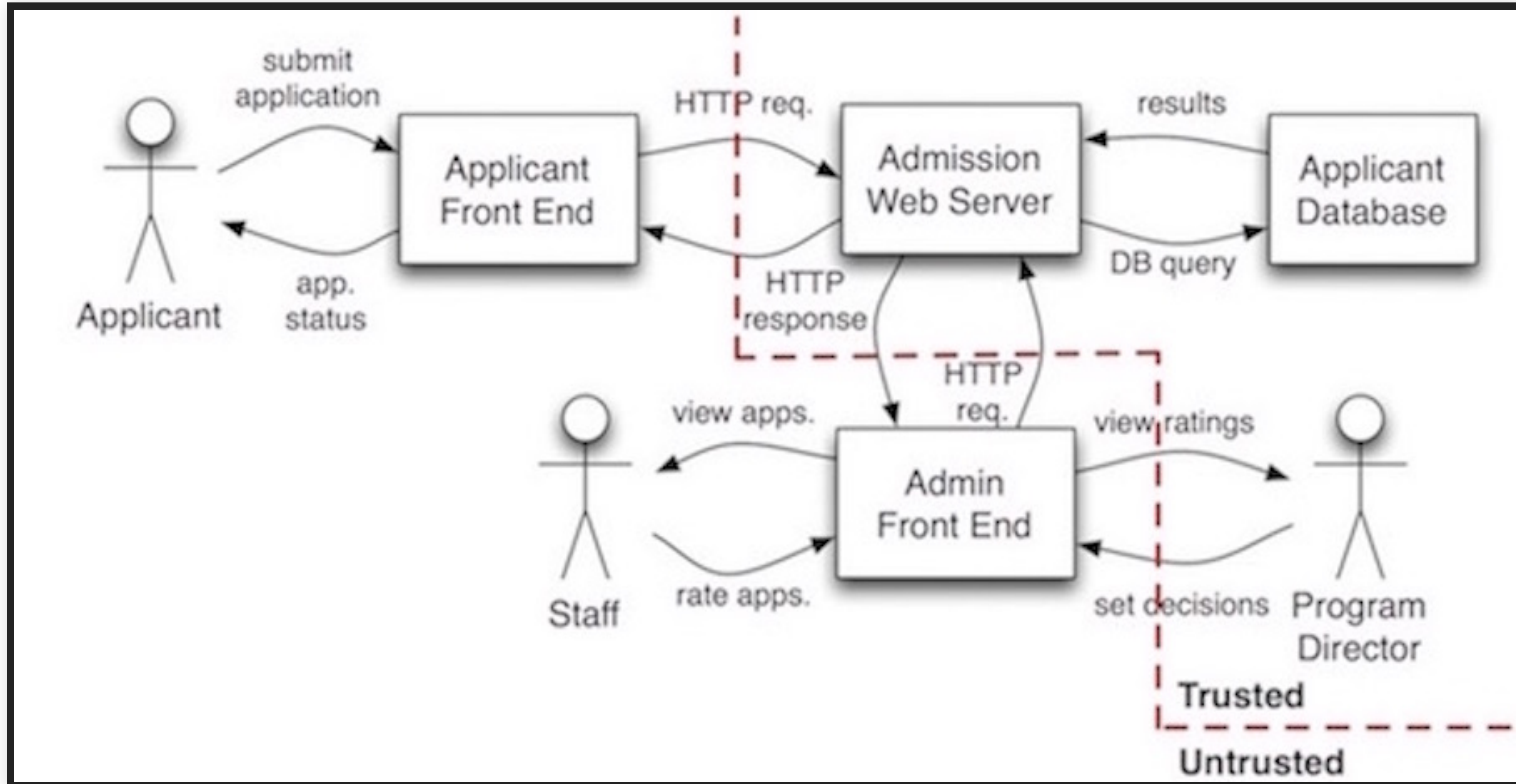
STRIDE THREAT MODELING

	Threat	Property Violated	Threat Definition
S	Spoofing identify	Authentication	Pretending to be something or someone other than yourself
T	Tampering with data	Integrity	Modifying something on disk, network, memory, or elsewhere
R	Repudiation	Non-repudiation	Claiming that you didn't do something or were not responsible; can be honest or false
I	Information disclosure	Confidentiality	Providing information to someone not authorized to access it
D	Denial of service	Availability	Exhausting resources needed to provide service
E	Elevation of privilege	Authorization	Allowing someone to do something they are not authorized to do

- A systematic approach to identifying threats (i.e., attacker actions)
 - Construct an architectural diagram with components & connections
 - Designate the trust boundary
 - For each untrusted component/connection, identify potential threats
 - For each potential threat, devise a mitigation strategy

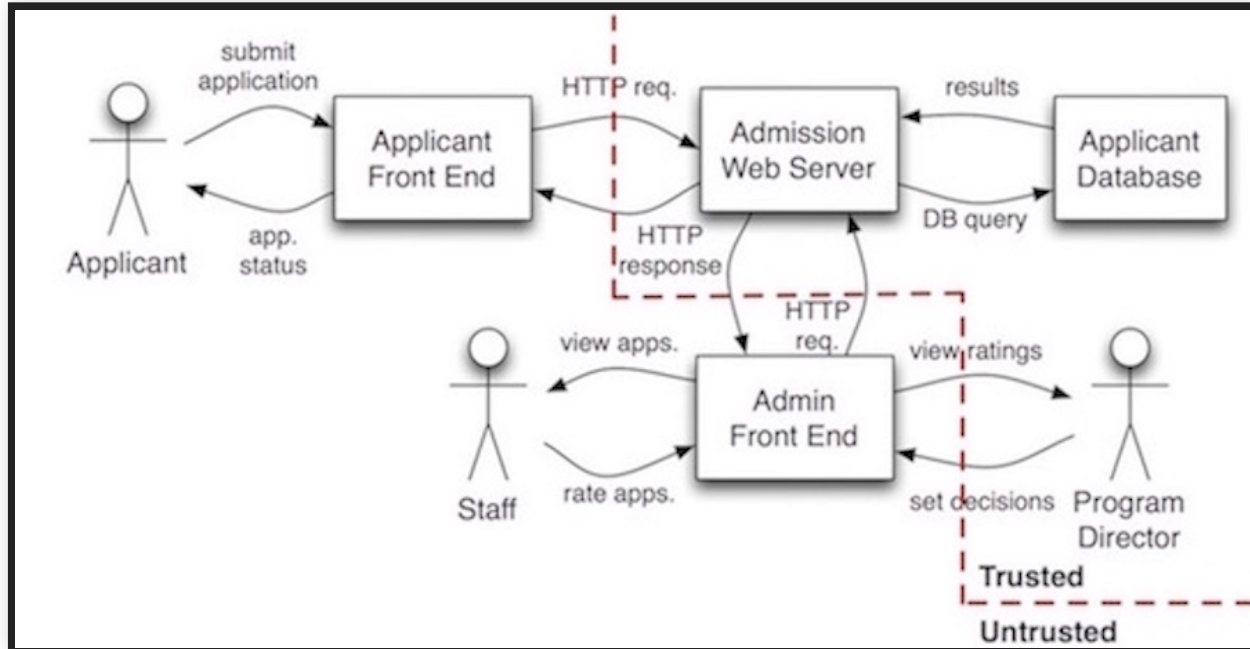
[More info: STRIDE approach](#)

STRIDE: COLLEGE ADMISSION



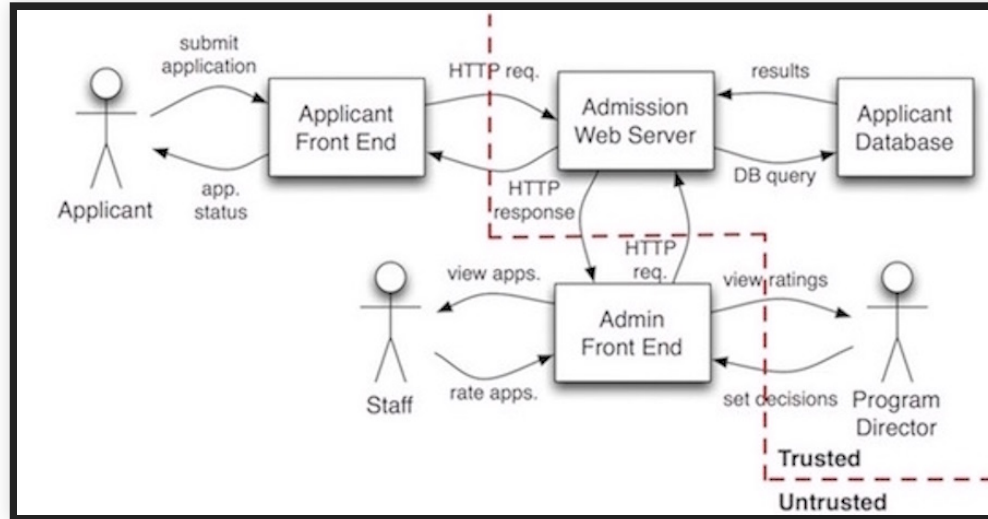
- Spoofing: ?
- Tampering: ?
- Information disclosure: ?
- Denial of service: ?

STRIDE: COLLEGE ADMISSION



- Spoofing: Attacker pretends to be another applicant by logging in
- Tampering: Attacker modifies applicant info using browser exploits
- Information disclosure: Attacker intercepts HTTP requests from/to server to read applicant info
- Denial of service: Attacker creates a large number of bogus accounts and overwhelms system with requests

STRIDE: MITIGATIONS



- Spoofing: Attacker pretends to be another applicant by logging in -> **Require stronger passwords**
- Tampering: Attacker modifies applicant info using browser exploits -> **Add server-side security tokens**
- Information disclosure: Attacker intercepts HTTP requests from/to server to read applicant info -> **Use encryption (HTTPS)**
- Denial of service: Attacker creates many bogus accounts and overwhelms system with requests -> **Limit requests per IP address**

STRIDE & OTHER THREAT MODELING METHODS

	Threat	Property Violated	Threat Definition
S	Spoofing identify	Authentication	Pretending to be something or someone other than yourself
T	Tampering with data	Integrity	Modifying something on disk, network, memory, or elsewhere
R	Repudiation	Non-repudiation	Claiming that you didn't do something or were not responsible; can be honest or false
I	Information disclosure	Confidentiality	Providing information to someone not authorized to access it
D	Denial of service	Availability	Exhausting resources needed to provide service
E	Elevation of privilege	Authorization	Allowing someone to do something they are not authorized to do

- A systematic approach to identifying threats & attacker actions
- Limitations:
 - May end up with a long list of threats, not all of them critical
 - False sense of security: STRIDE does not imply completeness!
- Consider cost vs. benefit trade-offs: Implementing mitigations add to development cost and complexity
 - Focus on most critical/likely threats

THREAT MODELING FOR ML

ML ATTACKER GOAL

ML ATTACKER GOAL

- Confidentiality attacks: Exposure of sensitive data

ML ATTACKER GOAL

- Confidentiality attacks: Exposure of sensitive data
 - Infer a sensitive label for a data point (e.g., hospital record)

ML ATTACKER GOAL

- Confidentiality attacks: Exposure of sensitive data
 - Infer a sensitive label for a data point (e.g., hospital record)
- Integrity attacks: Unauthorized modification of data

ML ATTACKER GOAL

- Confidentiality attacks: Exposure of sensitive data
 - Infer a sensitive label for a data point (e.g., hospital record)
- Integrity attacks: Unauthorized modification of data
 - Induce a model to misclassify data points from one class to another

ML ATTACKER GOAL

- Confidentiality attacks: Exposure of sensitive data
 - Infer a sensitive label for a data point (e.g., hospital record)
- Integrity attacks: Unauthorized modification of data
 - Induce a model to misclassify data points from one class to another
 - e.g., Spam filter: Classify a spam as a non-spam

ML ATTACKER GOAL

- Confidentiality attacks: Exposure of sensitive data
 - Infer a sensitive label for a data point (e.g., hospital record)
- Integrity attacks: Unauthorized modification of data
 - Induce a model to misclassify data points from one class to another
 - e.g., Spam filter: Classify a spam as a non-spam
- Availability attacks: Disruption to critical services

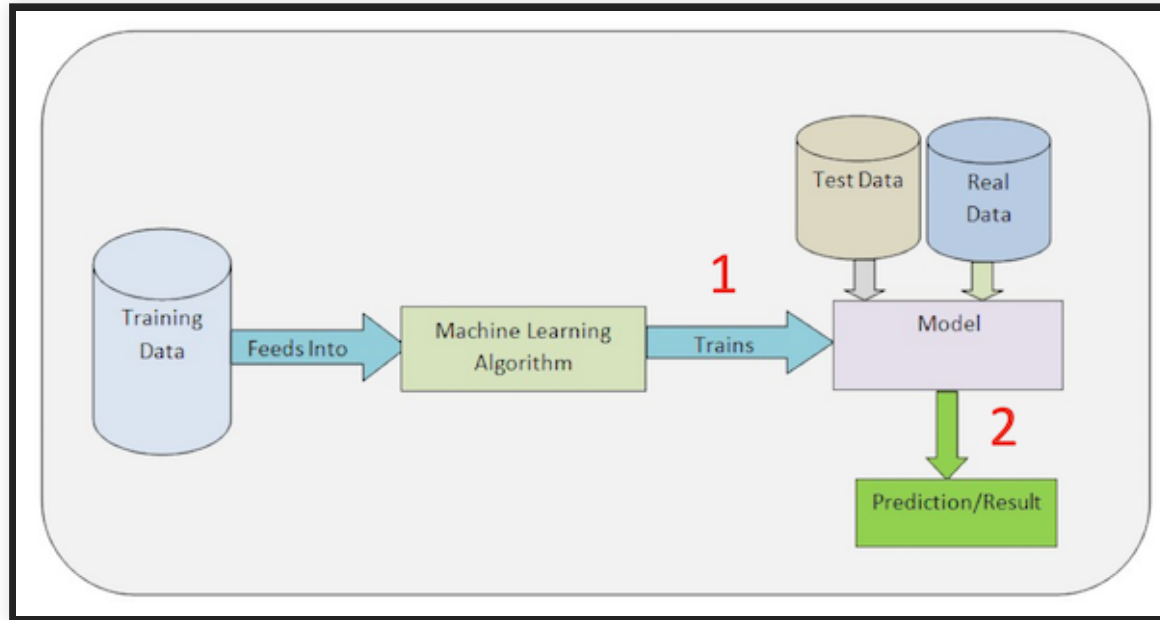
ML ATTACKER GOAL

- Confidentiality attacks: Exposure of sensitive data
 - Infer a sensitive label for a data point (e.g., hospital record)
- Integrity attacks: Unauthorized modification of data
 - Induce a model to misclassify data points from one class to another
 - e.g., Spam filter: Classify a spam as a non-spam
- Availability attacks: Disruption to critical services
 - Reduce the accuracy of a model

ML ATTACKER GOAL

- Confidentiality attacks: Exposure of sensitive data
 - Infer a sensitive label for a data point (e.g., hospital record)
- Integrity attacks: Unauthorized modification of data
 - Induce a model to misclassify data points from one class to another
 - e.g., Spam filter: Classify a spam as a non-spam
- Availability attacks: Disruption to critical services
 - Reduce the accuracy of a model
 - Induce a model to misclassify many data points

ML ATTACKER CAPABILITY

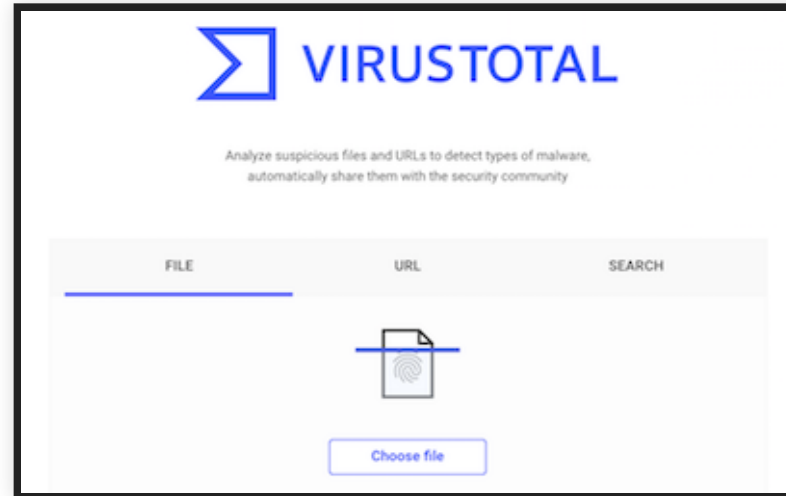


- Knowledge: Does the attacker have access to the model?
 - Training data? Learning algorithm used? Parameters?
- Attacker actions:
 - Training time: **Poisoning attacks**
 - Inference time: **Evasion attacks, model inversion attacks**

POISONING ATTACKS: AVAILABILITY



POISONING ATTACKS: AVAILABILITY



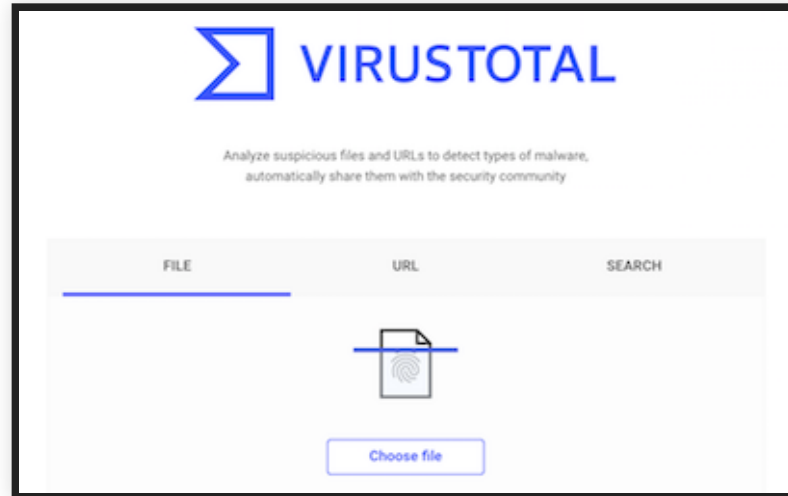
- Availability: Inject mislabeled training data to damage model quality
 - 3% poisoning => 11% decrease in accuracy (Steinhardt, 2017)

POISONING ATTACKS: AVAILABILITY



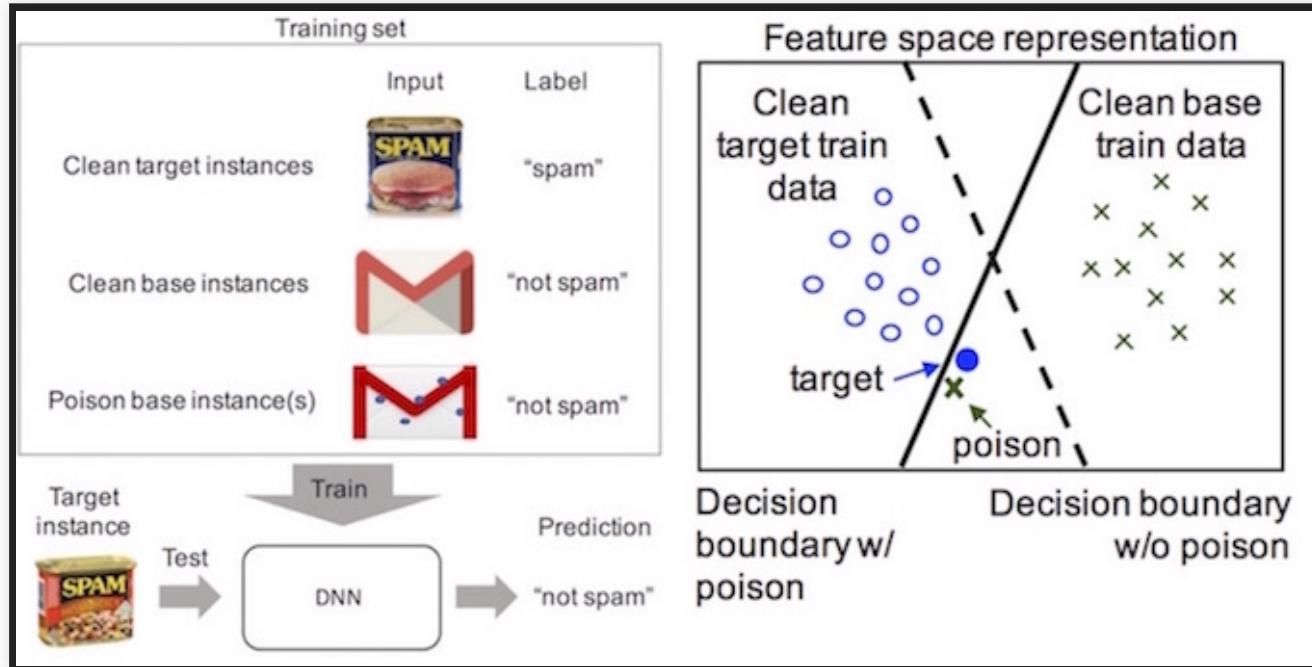
- Availability: Inject mislabeled training data to damage model quality
 - 3% poisoning => 11% decrease in accuracy (Steinhardt, 2017)
- Attacker must have some access to the training set
 - e.g., models trained on public data set (e.g., ImageNet)

POISONING ATTACKS: AVAILABILITY



- Availability: Inject mislabeled training data to damage model quality
 - 3% poisoning => 11% decrease in accuracy (Steinhardt, 2017)
- Attacker must have some access to the training set
 - e.g., models trained on public data set (e.g., ImageNet)
- Example: Anti-virus (AV) scanner
 - Online platform for submission of potentially malicious code
 - Some AV company (allegedly) poisoned competitor's model

POISONING ATTACKS: INTEGRITY



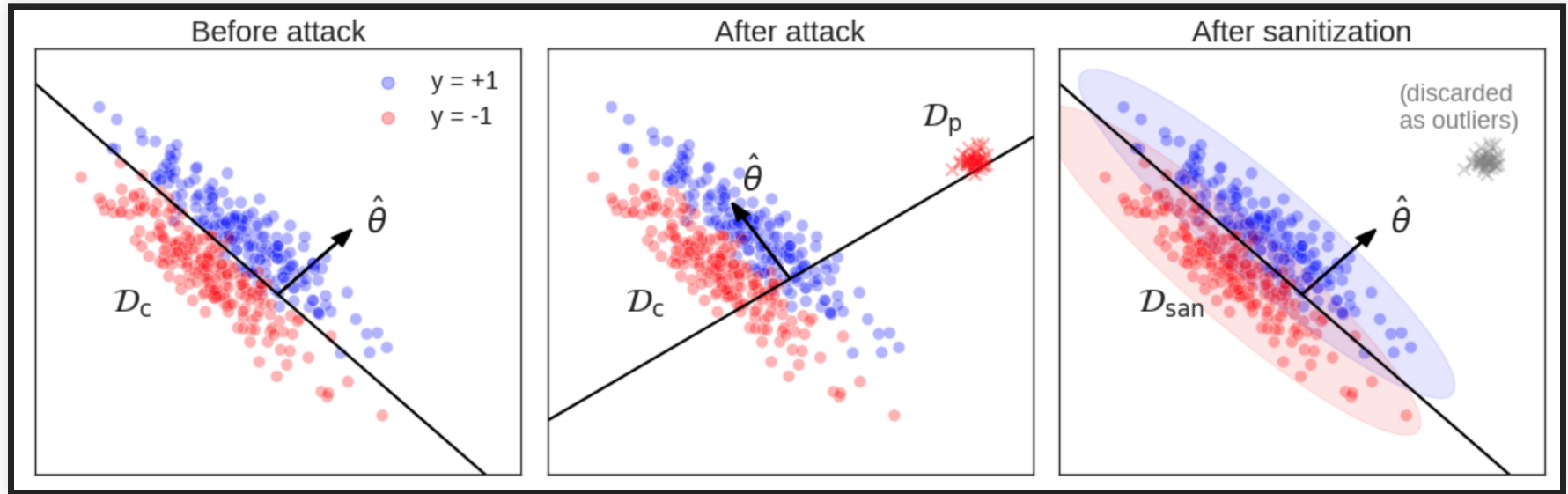
- Insert training data with seemingly correct labels
- More targeted than availability attacks
 - Cause misclassification from one specific class to another

Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks, Shafahi et al. (2018)

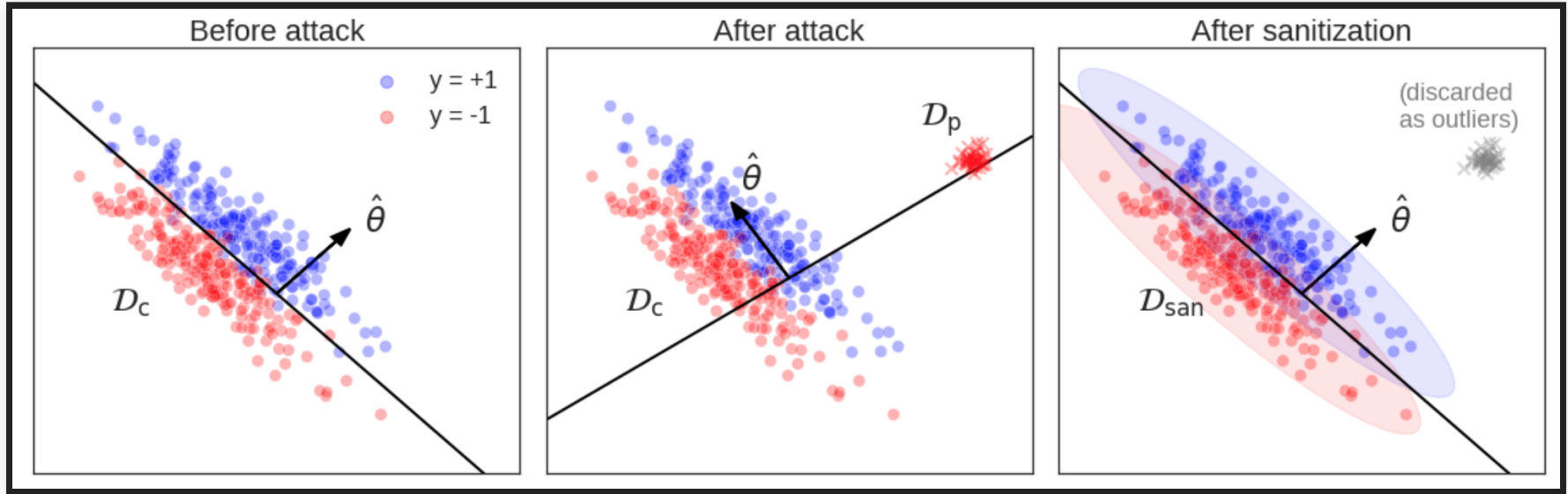
DEFENSE AGAINST POISONING ATTACKS

- Q. How would you mitigate poisoning attacks?

DEFENSE AGAINST POISONING ATTACKS

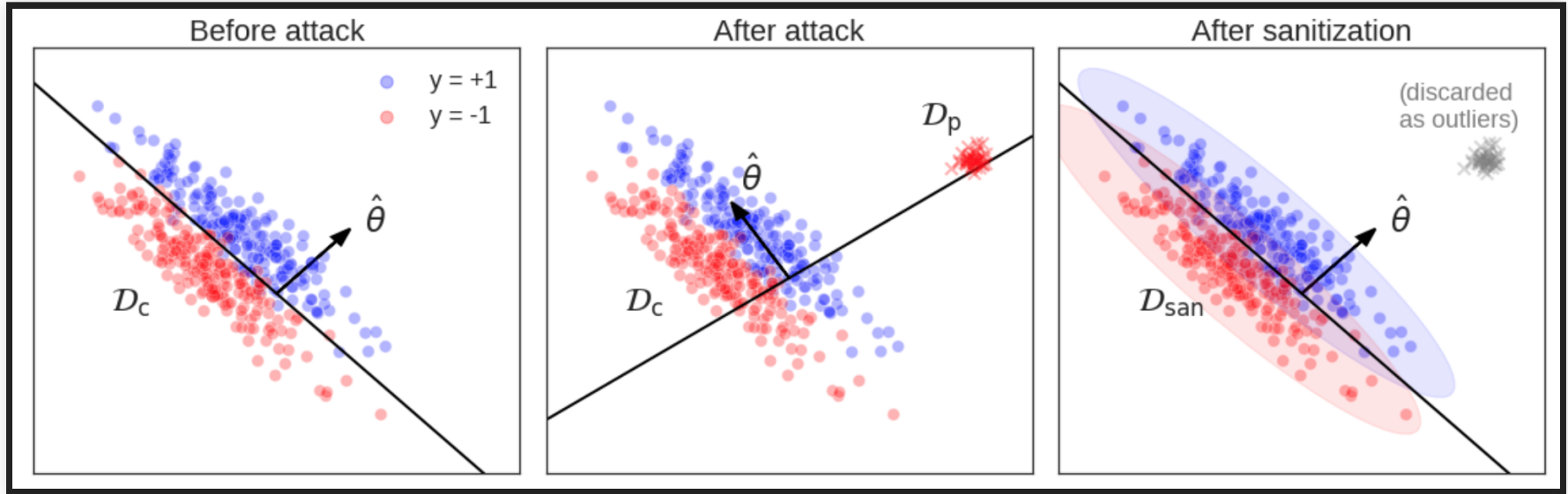


DEFENSE AGAINST POISONING ATTACKS



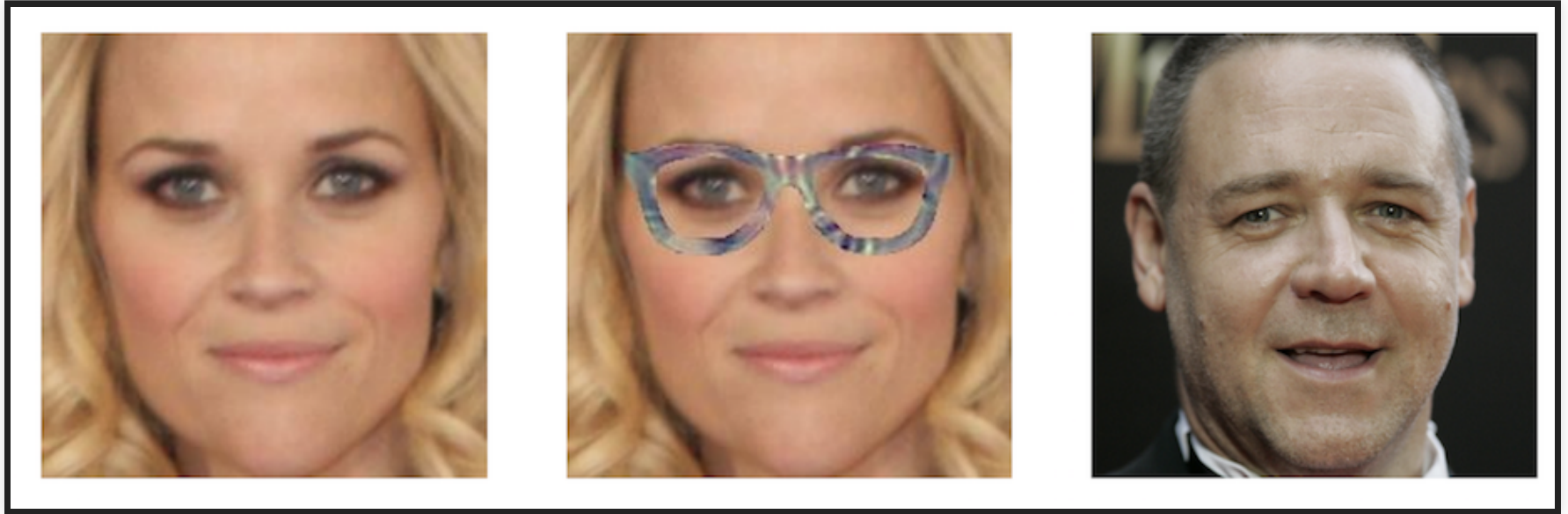
- Anomaly detection & data sanitization
 - Identify and remove outliers in training set (see [data quality lecture](#))
 - Identify and understand drift from telemetry

DEFENSE AGAINST POISONING ATTACKS



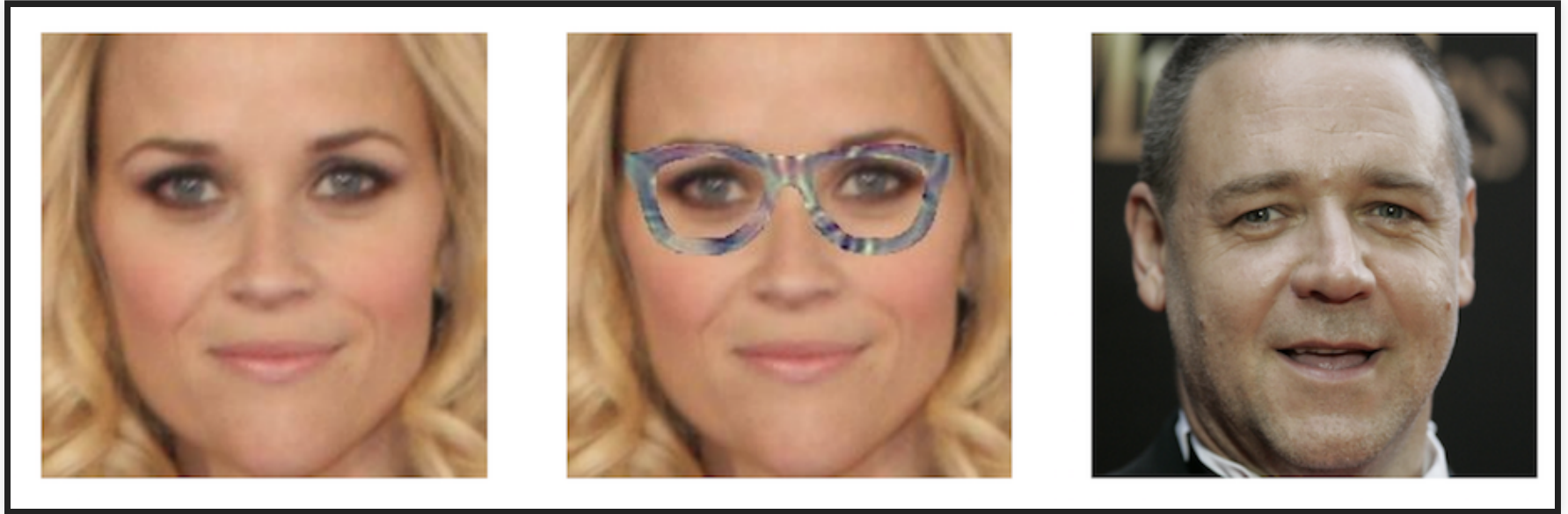
- Anomaly detection & data sanitization
 - Identify and remove outliers in training set (see [data quality lecture](#))
 - Identify and understand drift from telemetry
- Quality control over your training data
 - Who can modify or add to my training set? Do I trust the data source?
 - Use security mechanisms (e.g., authentication) and logging to track data provenance

EVASION ATTACKS (ADVERSARIAL EXAMPLES)



Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition, Sharif et al. (2016).

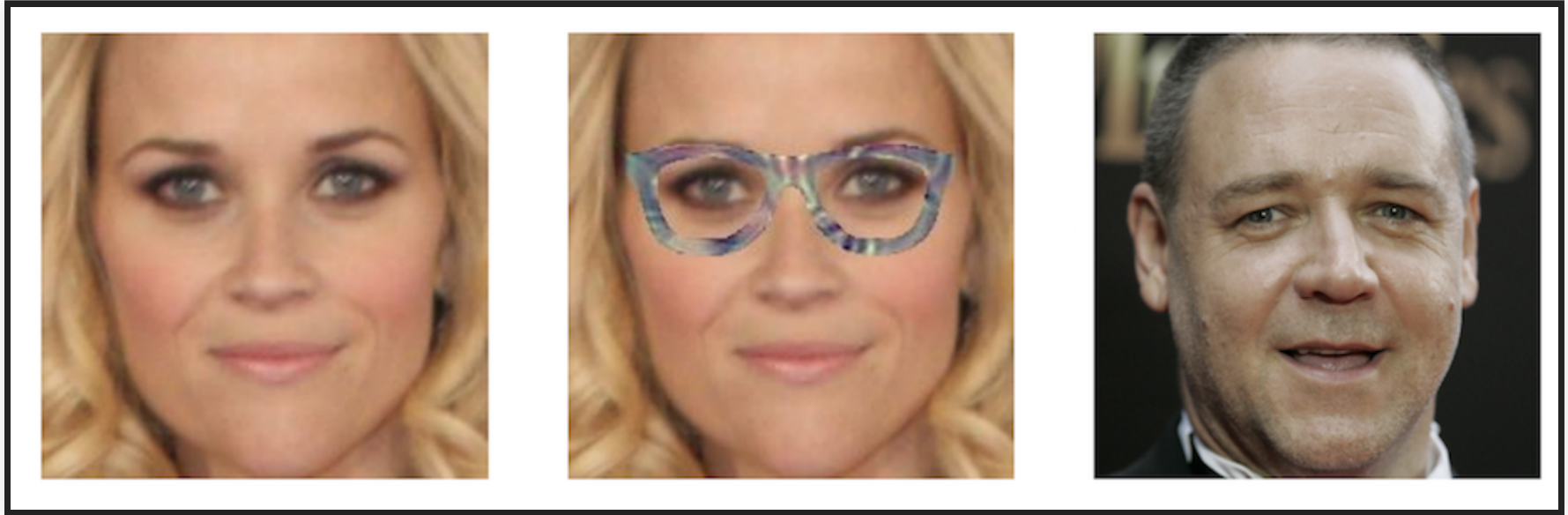
EVASION ATTACKS (ADVERSARIAL EXAMPLES)



- Add noise to an existing sample & cause misclassification

Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition, Sharif et al. (2016).

EVASION ATTACKS (ADVERSARIAL EXAMPLES)



- Add noise to an existing sample & cause misclassification
- Attack at inference time
 - Typically assumes knowledge of the model (algorithm, parameters)
 - Recently, shown to be possible even when the attacker only has access to model output ("blackbox" attack)

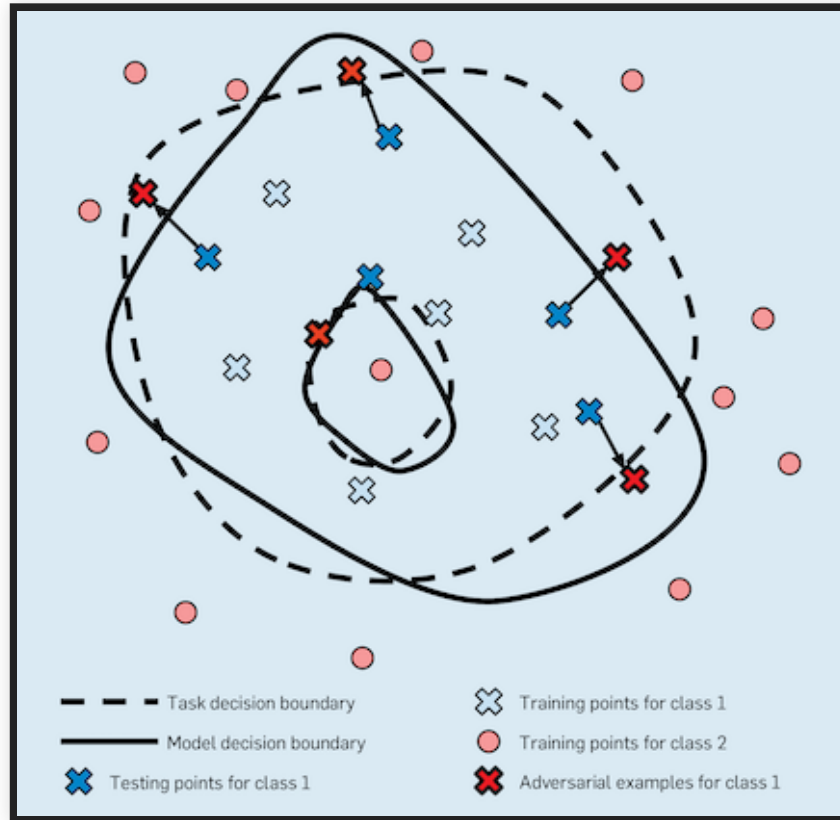
Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition, Sharif et al. (2016).

EVASION ATTACKS: ANOTHER EXAMPLE



Robust Physical-World Attacks on Deep Learning Visual Classification, Eykholt et al., in CVPR (2018).

TASK DECISION BOUNDARY VS MODEL BOUNDARY



- Decision boundary: Ground truth; often unknown and not specifiable
- Model boundary: What is learned; an *approximation* of decision boundary

From Goodfellow et al (2018). [Making machine learning robust against adversarial inputs](#). *Communications of the ACM*, 61(7), 56-66.

DEFENSE AGAINST EVASION ATTACKS

- Q. How would you mitigate evasion attacks?

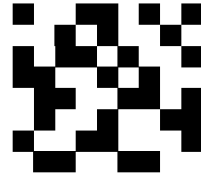
DEFENSE AGAINST EVASION ATTACKS



(a) Visual Image



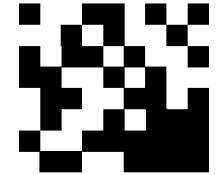
(b) Infrared Image of Smart Code



(c) Original Codeword

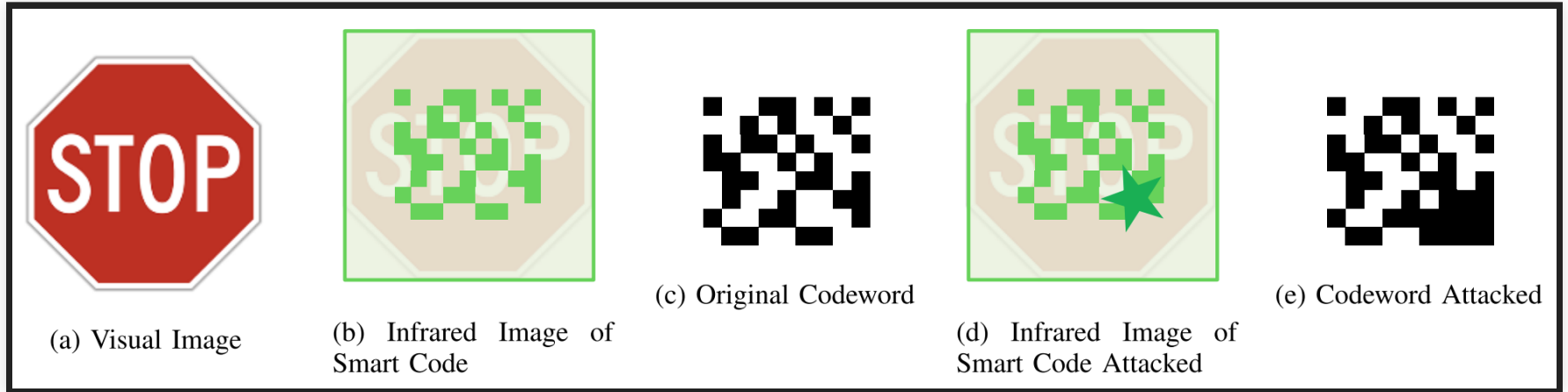


(d) Infrared Image of Smart Code Attacked



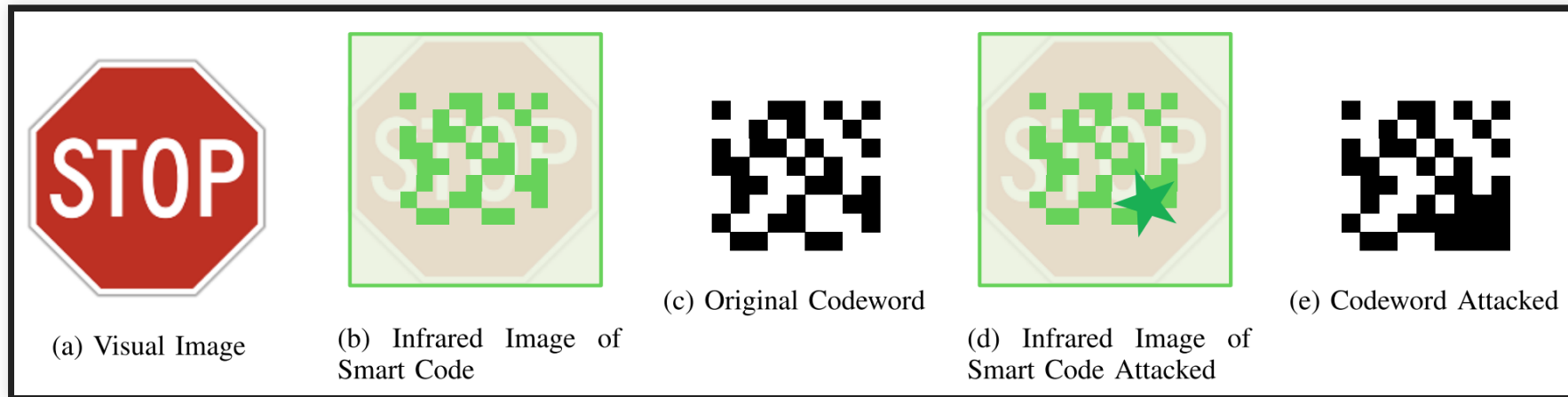
(e) Codeword Attacked

DEFENSE AGAINST EVASION ATTACKS



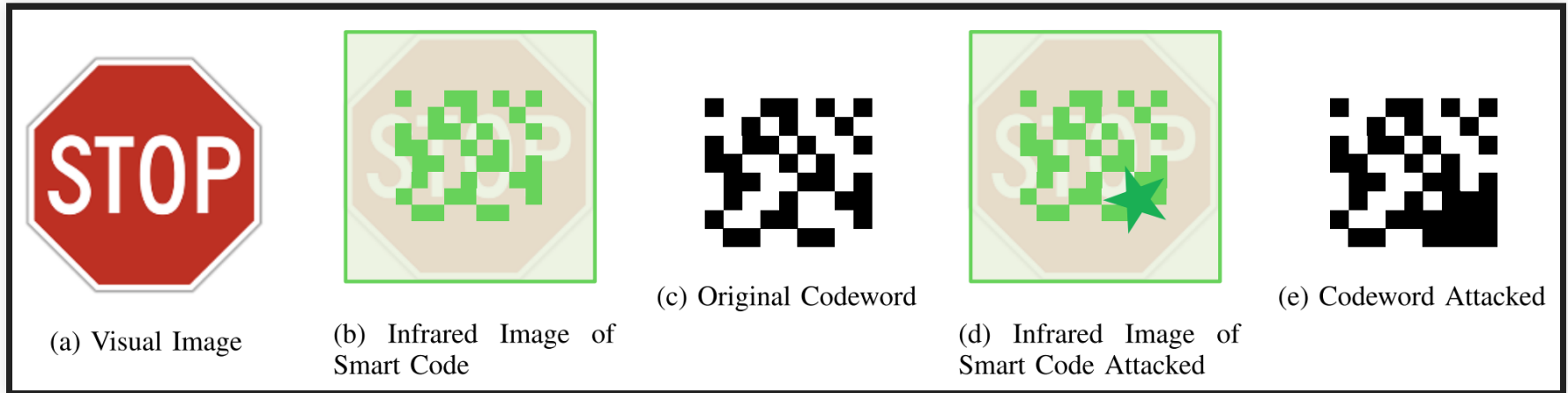
- Adversarial training
 - Generate/find a set of adversarial examples
 - Re-train your model with correct labels

DEFENSE AGAINST EVASION ATTACKS



- Adversarial training
 - Generate/find a set of adversarial examples
 - Re-train your model with correct labels
- Input sanitization
 - "Clean" & remove noise from input samples
 - e.g., Color depth reduction, spatial smoothing, JPEG compression

DEFENSE AGAINST EVASION ATTACKS



- Adversarial training
 - Generate/find a set of adversarial examples
 - Re-train your model with correct labels
- Input sanitization
 - "Clean" & remove noise from input samples
 - e.g., Color depth reduction, spatial smoothing, JPEG compression
- Redundancy: Design multiple mechanisms to detect an attack
 - Stop sign: Insert a barcode as a checksum; harder to bypass

GENERATING ADVERSARIAL EXAMPLES

- Q. How do we generate adversarial examples?

GENERATING ADVERSARIAL EXAMPLES

- See [counterfactual explanations](#)
- Find similar inputs with different predictions
 - Can be targeted (specific prediction) or untargeted (any wrong prediction)
- Many similarity measures (e.g., change one feature vs small changes to many features)
 - $x^* = x + \operatorname{argmin}\{|\epsilon| : f(x + \epsilon) \neq f(x)\}$
- Attacks more effective with access to model internals, but black-box attacks also feasible
 - With model internals: Follow the model's gradient
 - Without model internals: Learn [surrogate model](#)
 - With access to confidence scores: Heuristic search (e.g., hill climbing)

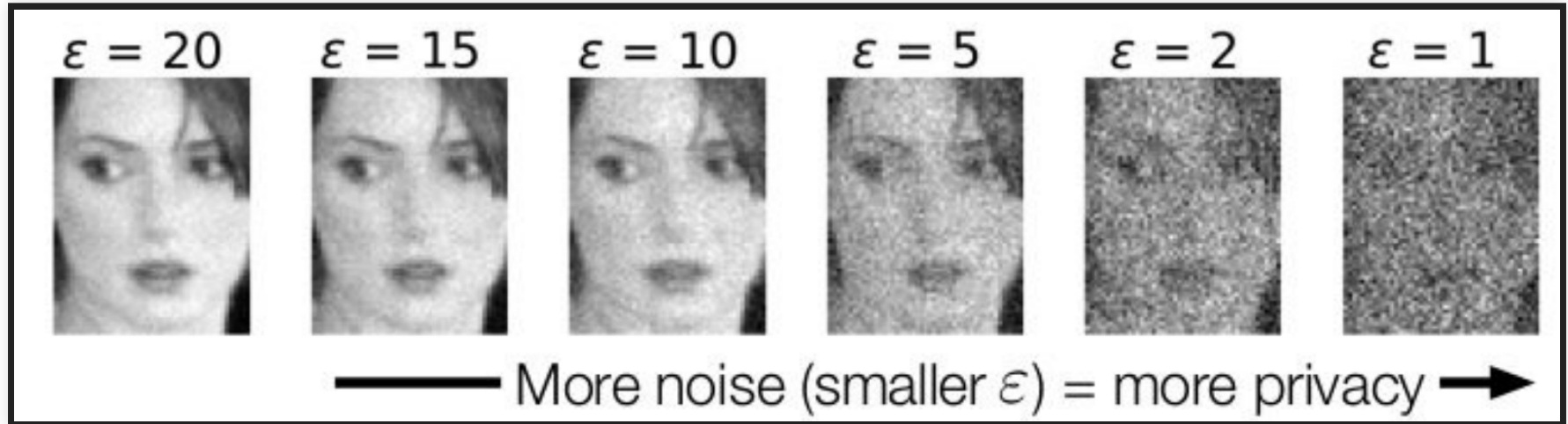
MODEL INVERSION: CONFIDENTIALITY



- Given a model output (e.g., name of a person), infer the corresponding, potentially sensitive input (facial image of the person)
- One method: Gradient descent on input space
 - Assumes that the model produces a confidence score for prediction
 - Start with a random input vector & iterate towards input values with higher confidence level

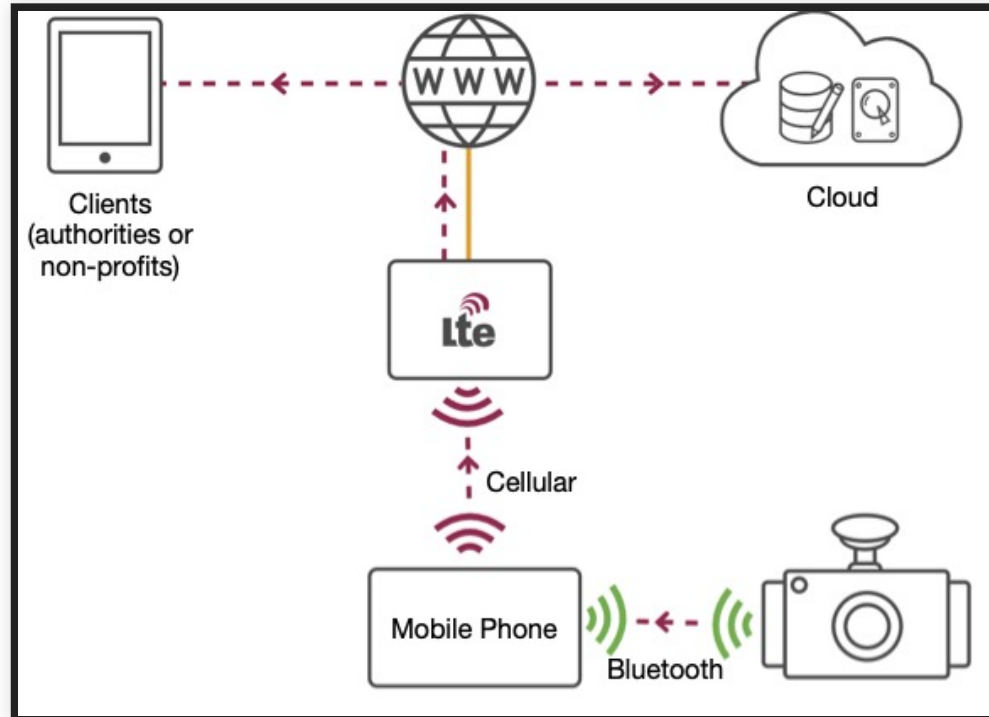
Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures, M. Fredrikson et al. in CCS (2015).

DEFENSE AGAINST MODEL INVERSION ATTACKS



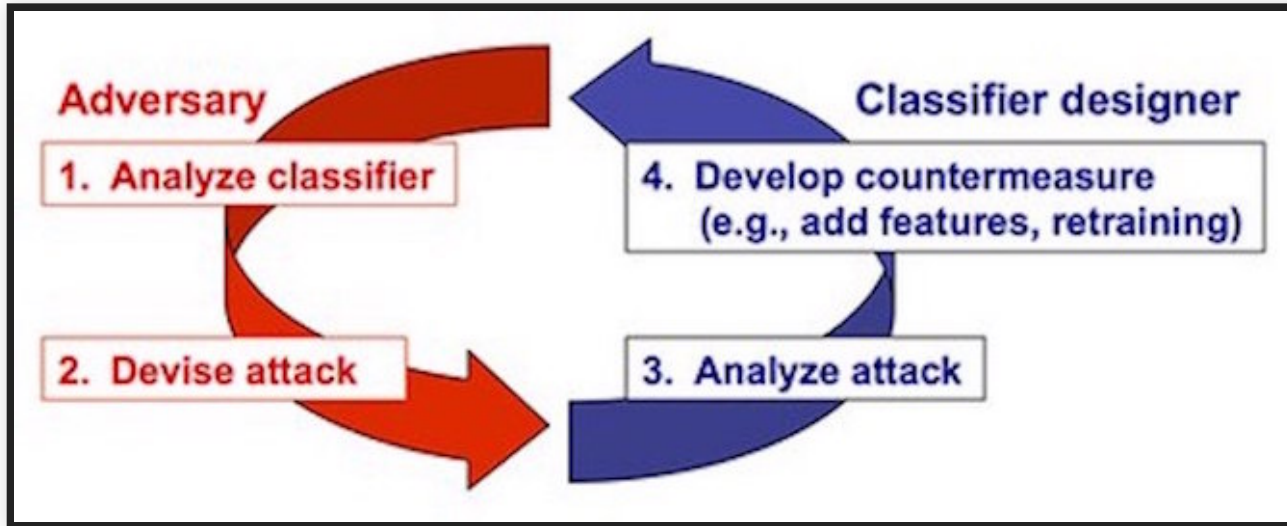
- Limit attacker access to confidence scores
 - e.g., reduce the precision of the scores by rounding them off
 - But also reduces the utility of legitimate use of these scores!
- Differential privacy in ML
 - Limit what attacker can learn about the model (e.g., parameters) based on an individual training sample
 - Achieved by adding noise to input or output (e.g., DP-SGD)
 - More noise => higher privacy, but also lower model accuracy!

BREAKOUT: DASHCAM SYSTEM

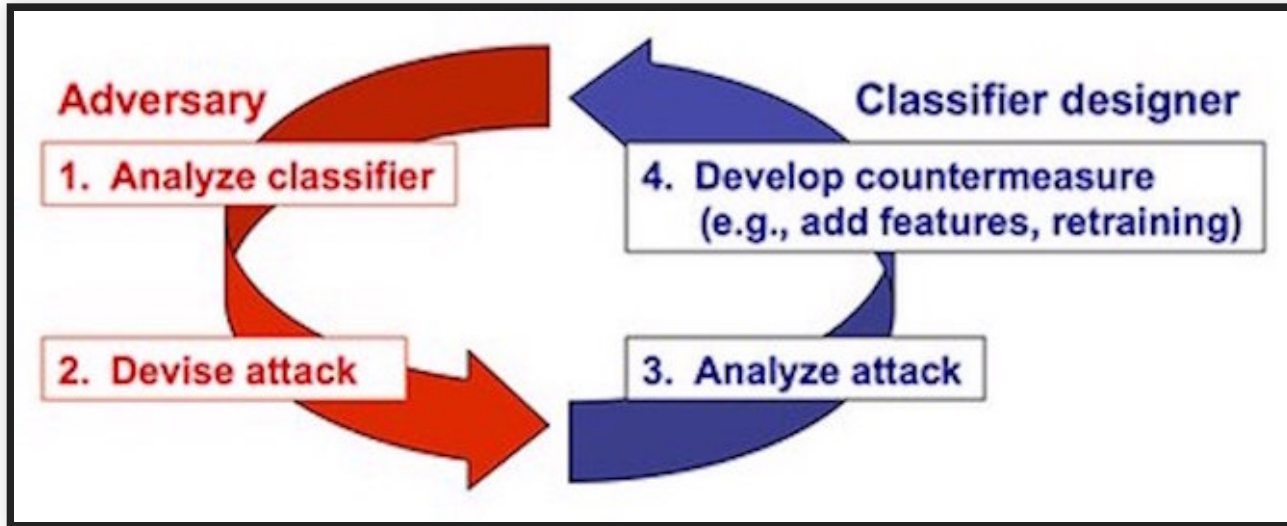


- Recall: Dashcam system from I2
- Post on #lecture in Slack:
 - What are the security requirements?
 - What are possible (ML) attacks on the system?
 - What are some possible mitigations against these attacks?

STATE OF ML SECURITY

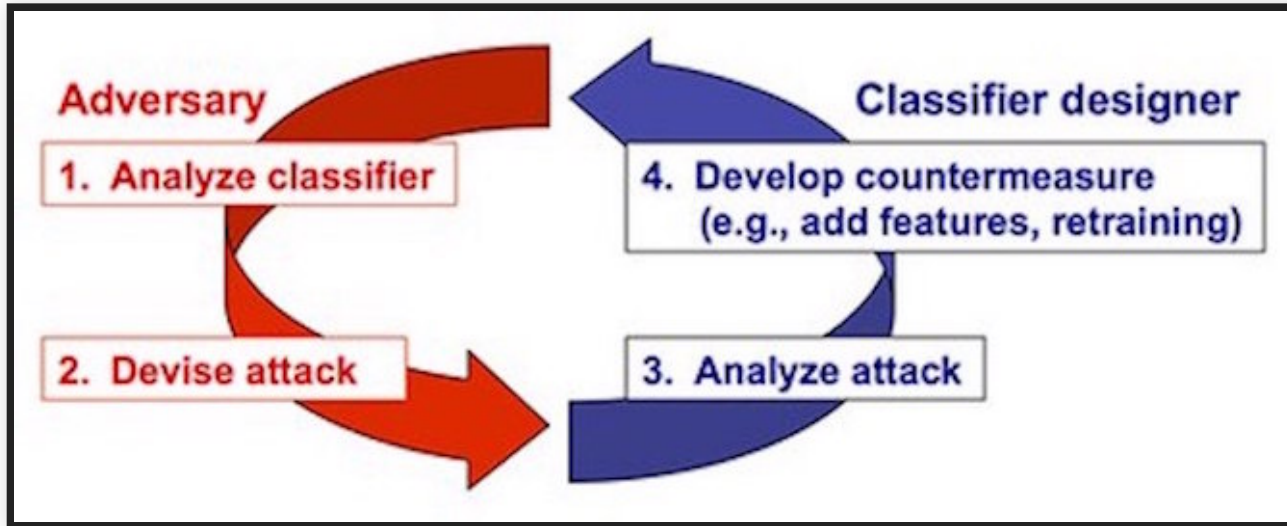


STATE OF ML SECURITY



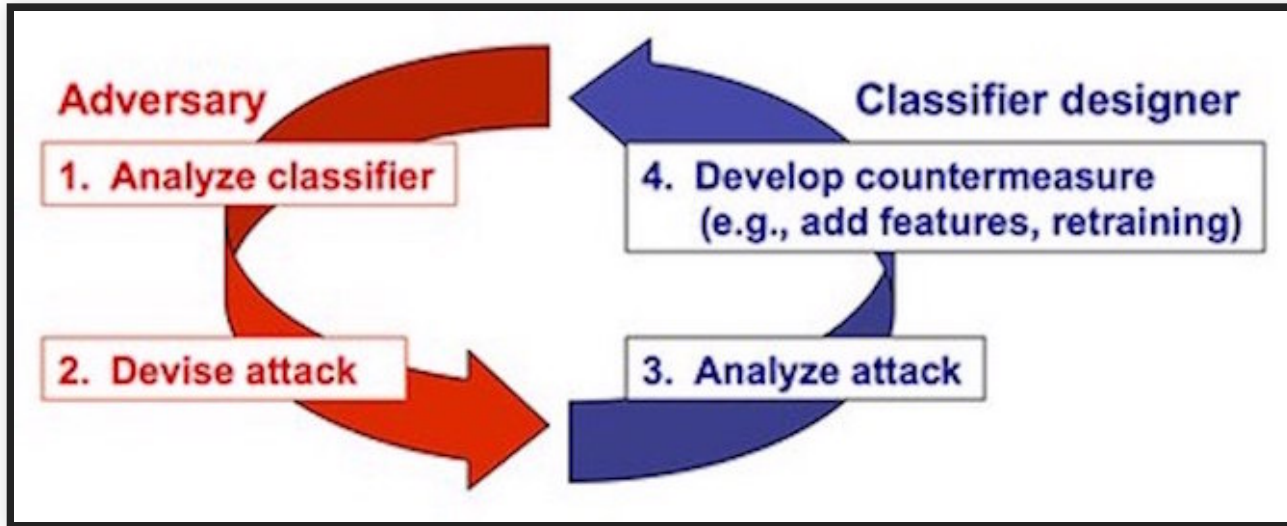
- On-going arms race (mostly among researchers)
 - Defenses proposed & quickly broken by noble attacks

STATE OF ML SECURITY



- On-going arms race (mostly among researchers)
 - Defenses proposed & quickly broken by noble attacks
- Assume ML component is likely vulnerable
 - Design your system to minimize impact of an attack

STATE OF ML SECURITY



- On-going arms race (mostly among researchers)
 - Defenses proposed & quickly broken by noble attacks
- Assume ML component is likely vulnerable
 - Design your system to minimize impact of an attack
- Remember: There may be easier ways to compromise system
 - e.g., poor security misconfiguration (default password), lack of encryption, code vulnerabilities, etc.,

DESIGNING FOR SECURITY

SECURITY MINDSET



- Assume that all components may be compromised at one point or another
- Don't assume users will behave as expected; assume all inputs to the system as potentially malicious
- Aim for risk minimization, not perfect security; reduce the chance of catastrophic failures from attacks

SECURE DESIGN PRINCIPLES

SECURE DESIGN PRINCIPLES

- Goal: Minimize the impact of a compromised component on the rest of the system
 - In poor system designs, vulnerability in one component => entire system compromised!

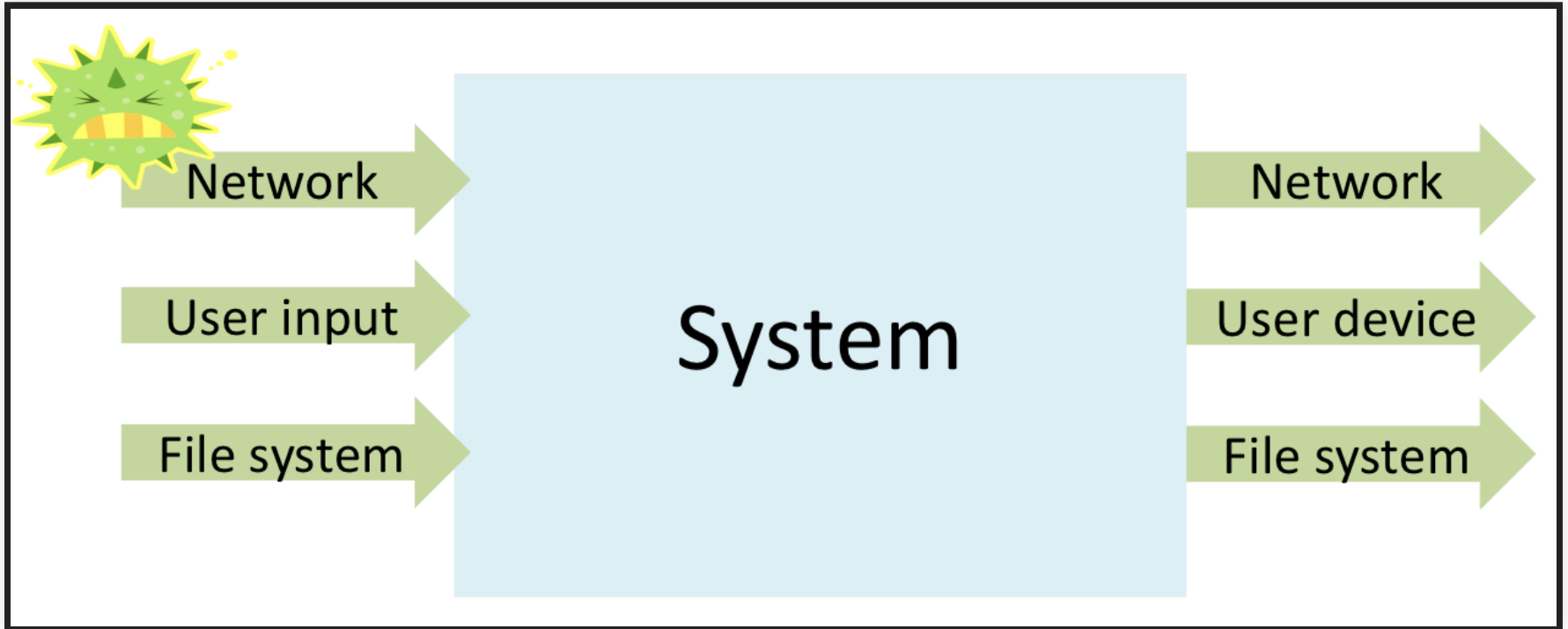
SECURE DESIGN PRINCIPLES

- Goal: Minimize the impact of a compromised component on the rest of the system
 - In poor system designs, vulnerability in one component => entire system compromised!
- Principle of least privilege
 - A component should be given the minimal privileges needed to fulfill its functionality

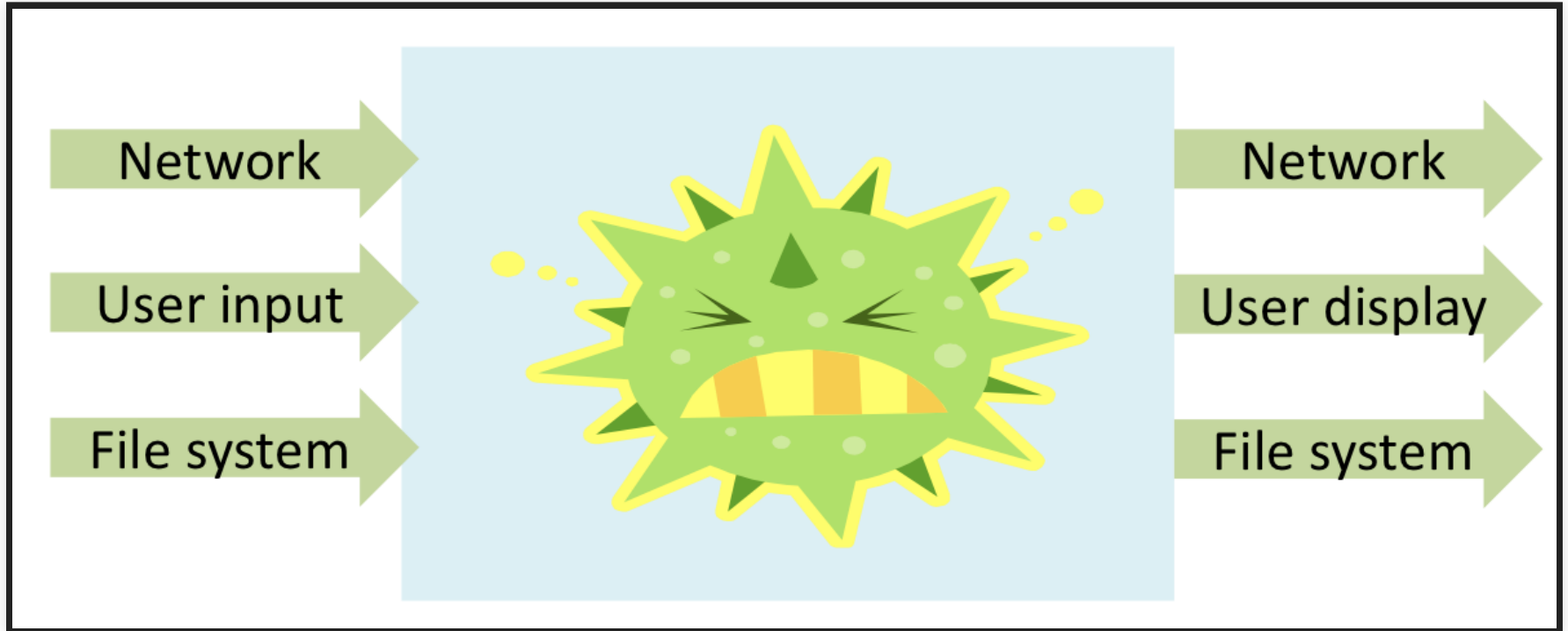
SECURE DESIGN PRINCIPLES

- Goal: Minimize the impact of a compromised component on the rest of the system
 - In poor system designs, vulnerability in one component => entire system compromised!
- Principle of least privilege
 - A component should be given the minimal privileges needed to fulfill its functionality
- Isolation/compartmentalization
 - Components should be able to interact with each other no more than necessary
 - Components should treat inputs from each other as potentially malicious

MONOLITHIC DESIGN

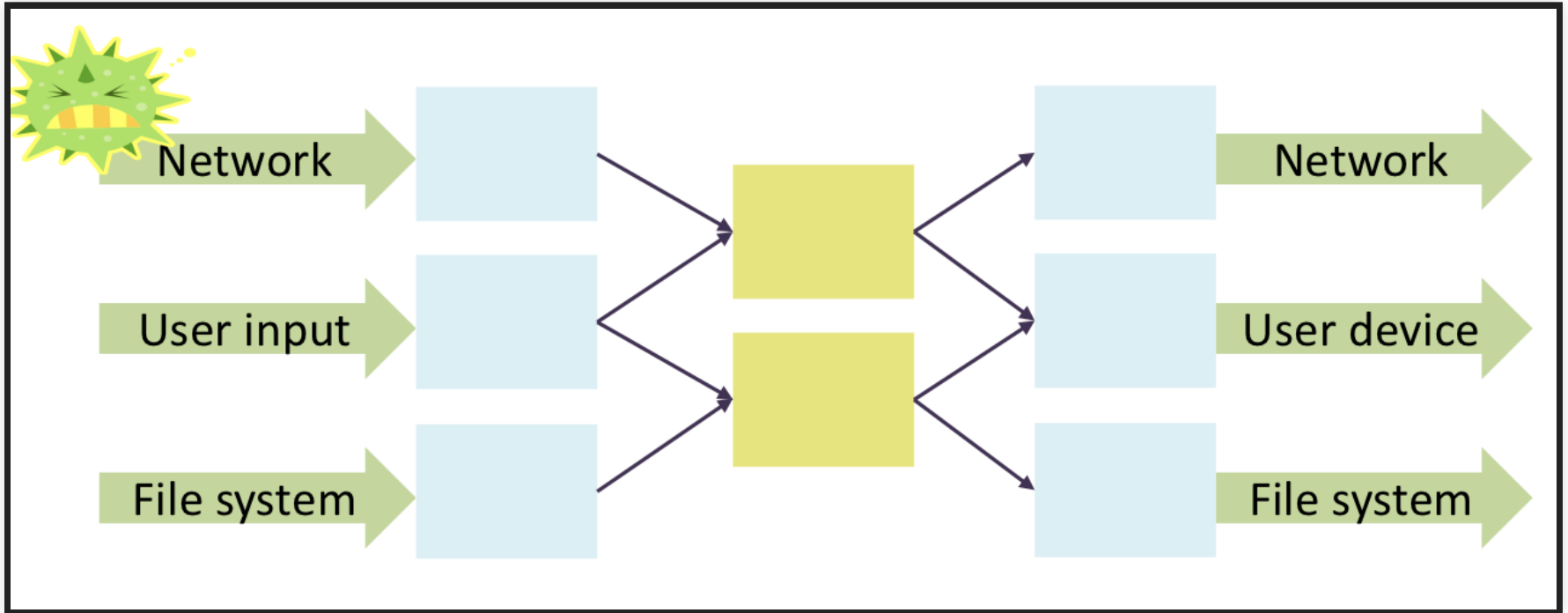


MONOLITHIC DESIGN

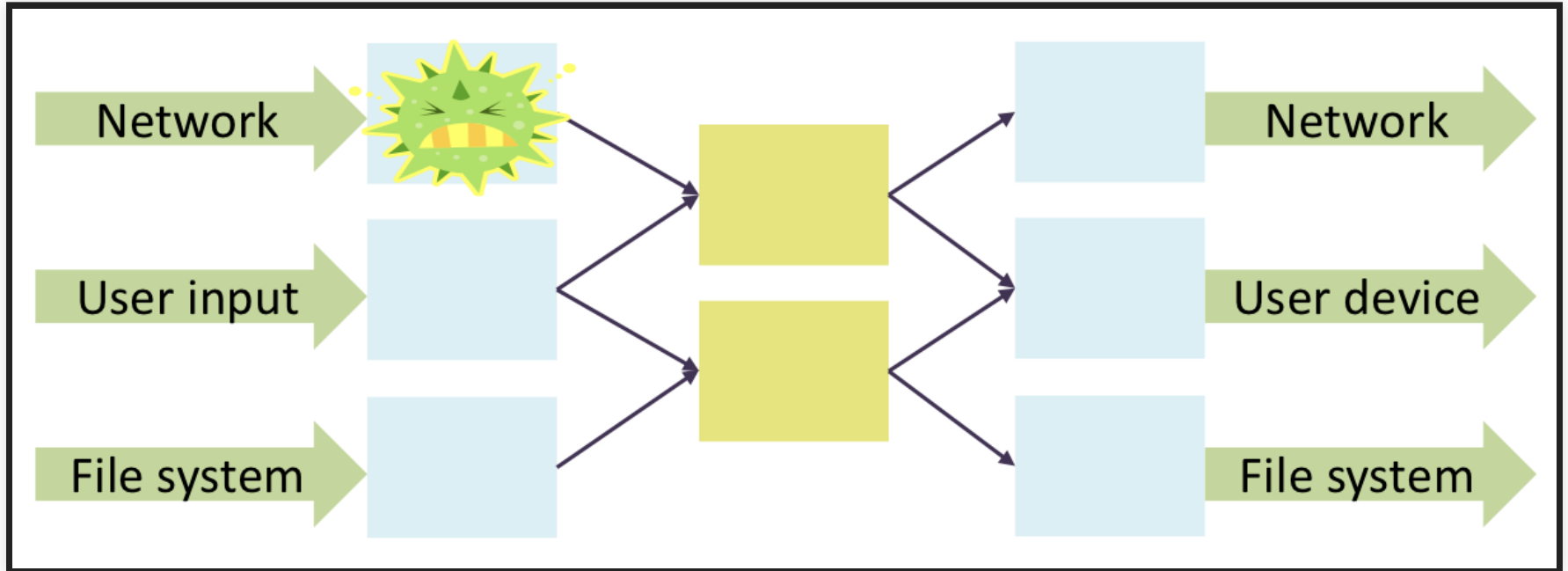


Flaw in any part of the system => Security impact on the entire system!

COMPARTMENTALIZED DESIGN

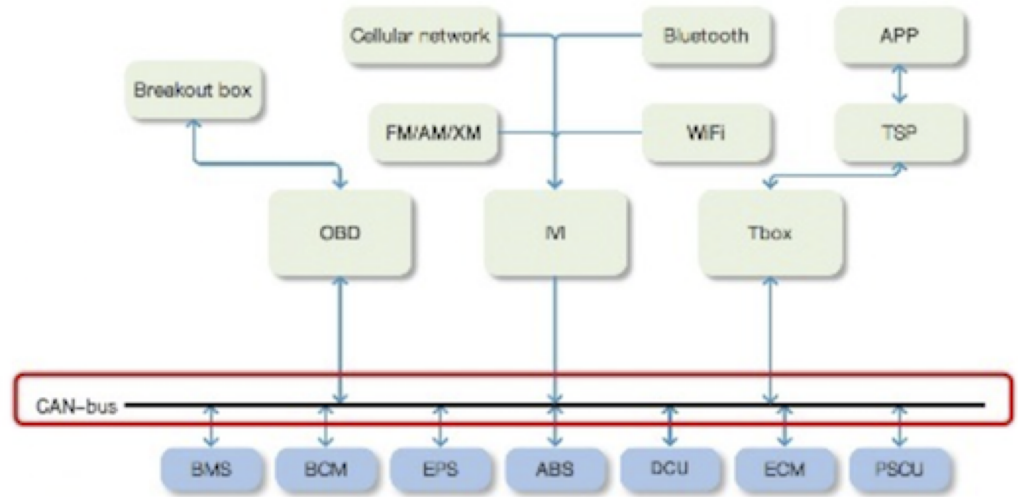


COMPARTMENTALIZED DESIGN



Flaw in one component => Limited impact on the rest of the system!

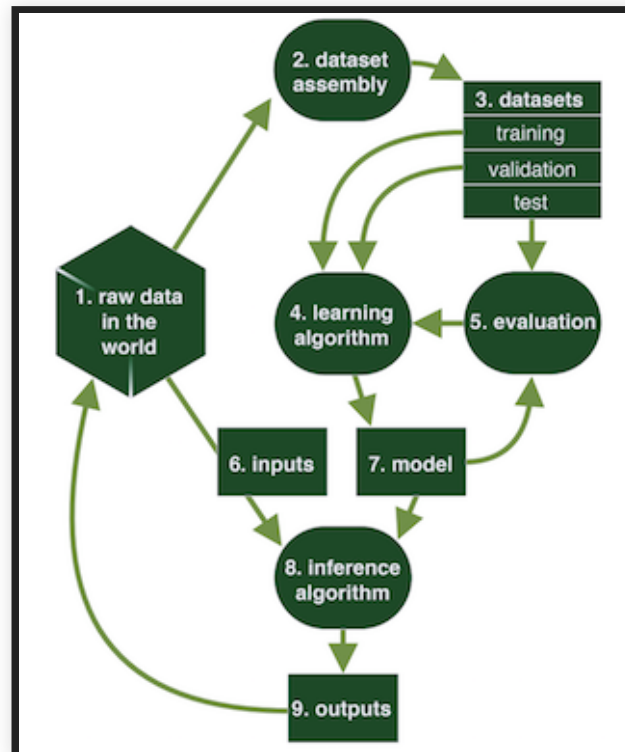
EXAMPLE: VEHICLE SECURITY



- Research project from UCSD: Remotely taking over vehicle control
 - Create MP3 with malicious code & burn onto CD
 - Play CD => send malicious commands to brakes, engine, locks...
- Problem: Over-privilege & lack of isolation!
 - In traditional vehicles, components share a common CAN bus
 - Anyone can broadcast & read messages

SECURE DESIGN PRINCIPLES FOR ML

- Principle of least privilege
 - Who has access to training data, model internal, system input & output, etc.,?
 - Does any user/stakeholder have more access than necessary?
 - If so, limit access by using authentication mechanisms



SECURE DESIGN PRINCIPLES FOR ML

- Principle of least privilege
 - Who has access to training data, model internal, system input & output, etc.,?
 - Does any user/stakeholder have more access than necessary?
 - If so, limit access by using authentication mechanisms

SECURE DESIGN PRINCIPLES FOR ML

- Principle of least privilege
 - Who has access to training data, model internal, system input & output, etc.,?
 - Does any user/stakeholder have more access than necessary?
 - If so, limit access by using authentication mechanisms
- Isolation & compartmentalization
 - Can a security attack on one ML component (e.g., misclassification) adversely affect other parts of the system?
 - If so, compartmentalize or build in mechanisms to limit impact (see [risk mitigation strategies](#))

SECURE DESIGN PRINCIPLES FOR ML

- Principle of least privilege
 - Who has access to training data, model internal, system input & output, etc.,?
 - Does any user/stakeholder have more access than necessary?
 - If so, limit access by using authentication mechanisms
- Isolation & compartmentalization
 - Can a security attack on one ML component (e.g., misclassification) adversely affect other parts of the system?
 - If so, compartmentalize or build in mechanisms to limit impact (see [risk mitigation strategies](#))
- Monitoring & detection:
 - Look for odd shifts in the dataset and clean the data if needed (for poisoning attacks)
 - Assume all system input as potentially malicious & sanitize (evasion attacks)

AI FOR SECURITY



30 COMPANIES MERGING AI AND CYBERSECURITY TO KEEP US SAFE AND SOUND

Alyssa Schroer

July 12, 2019 Updated: July 15, 2020

By the year 2021, cybercrime losses will

MANY DEFENSE SYSTEMS USE MACHINE LEARNING

- Classifiers to learn malicious content
 - Spam filters, virus detection
- Anomaly detection
 - Identify unusual/suspicious activity, eg. credit card fraud, intrusion detection
 - Often unsupervised learning, e.g. clustering
- Game theory
 - Model attacker costs and reactions, design countermeasures
- Automate incidence response and mitigation activities
 - Integrated with DevOps
- Network analysis
 - Identify bad actors and their communication in public/intelligence data
- Many more, huge commercial interest

Recommended reading: Chandola, Varun, Arindam Banerjee, and Vipin Kumar. "[Anomaly detection: A survey.](#)" ACM computing surveys (CSUR) 41, no. 3 (2009): 1-58.

AI SECURITY SOLUTIONS ARE AI-ENABLED SYSTEMS TOO

- AI/ML component one part of a larger system
- Consider entire system, from training to telemetry, to user interface, to pipeline automation, to monitoring
- AI-based security solutions can be attacked themselves

EQUIFAX

Speaker notes

One contributing factor to the Equifax attack was an expired certificate for an intrusion detection system

ML & DATA PRIVACY

TECH

How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did

Kashmir Hill Former Staff

Welcome to The Not-So Private Parts where technology & privacy collide

Follow

Andrew Pole, who heads a 60-person team at Target that studies customer behavior, boasted at a conference in 2010 about a proprietary program that could identify women - based on their purchases and demographic profile - who were pregnant.

<https://www.reuters.com/article/us-target-breach-datamining/what-target-knows-about-you-idUSBREA0M1JM20140123>

What Does Big Tech Know About You?

Basically Everything

Security Baron examined the privacy policies of Facebook, Google, Apple, Twitter, Amazon, and Microsoft; just how much these tech giants actually know about you might be surprising..



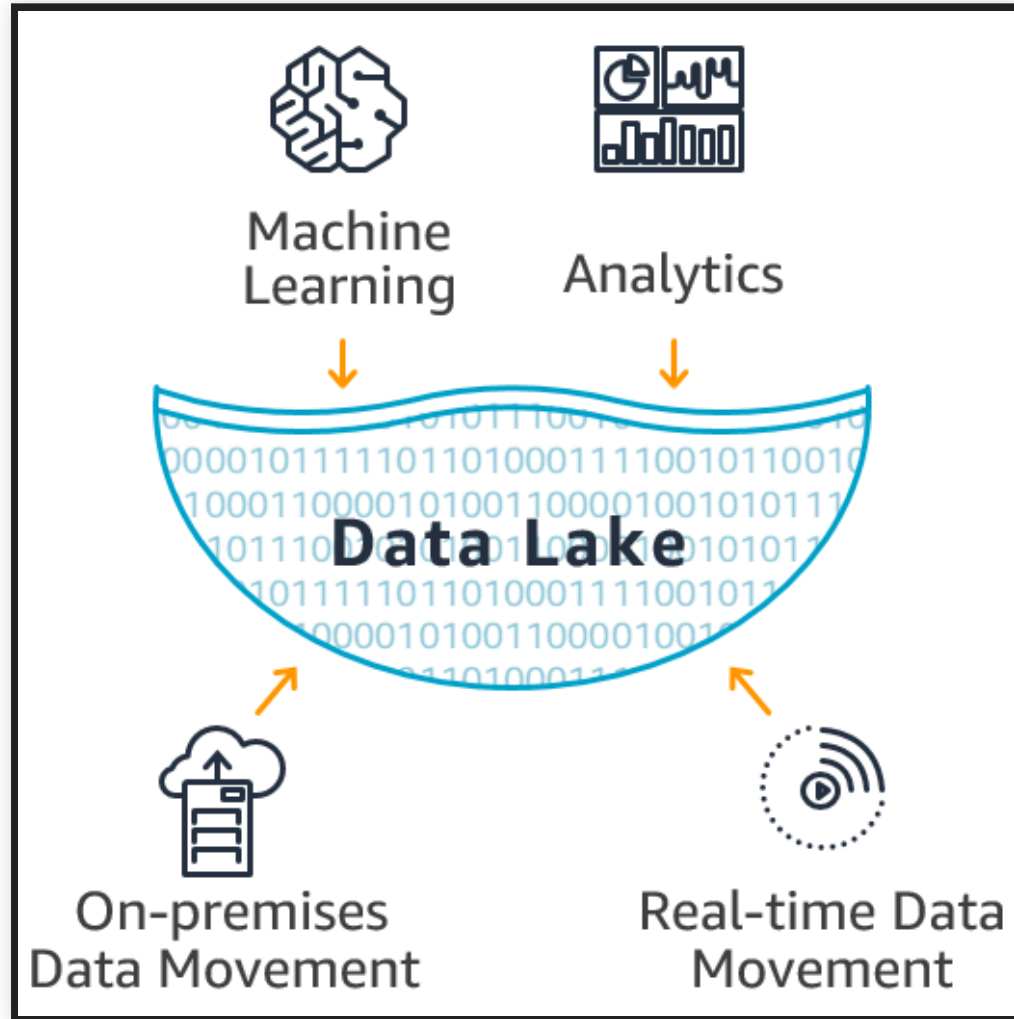
By [Angela Moscaritolo](#)

Updated January 18, 2022



		Google	Facebook	Apple	Twitter	Amazon	Microsoft
Name					×		
Gender				×	×	×	
Birthday				×	×	×	
Phone Number							
Email Address							

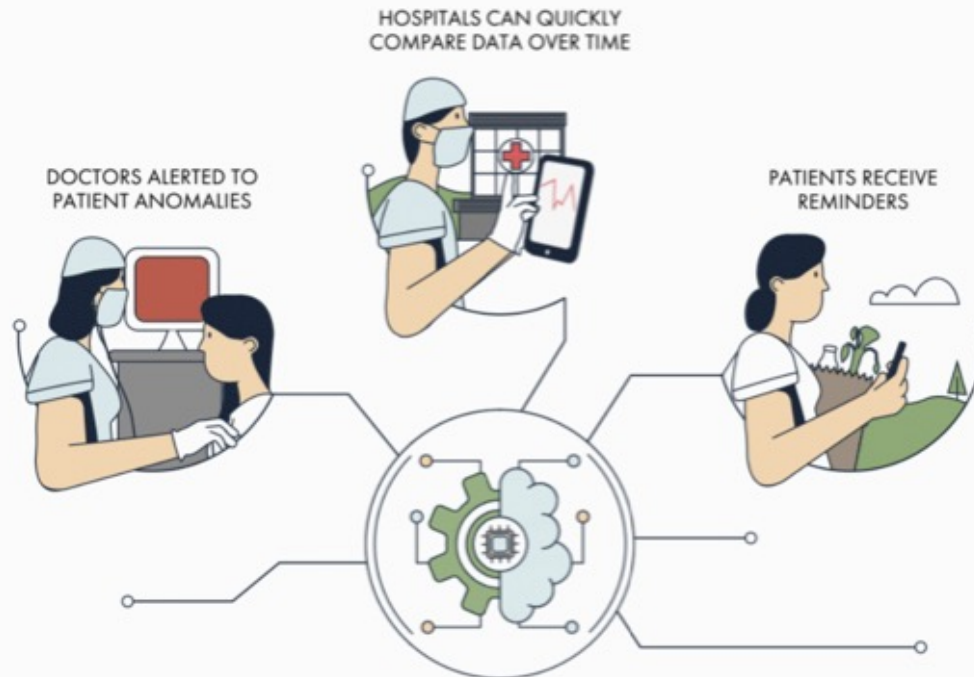
DATA LAKES



DATA PRIVACY VS UTILITY

PARTNER CONTENT WIRED INSIDER

FROM DIAGNOSIS TO HOLISTIC PATIENT CARE, MACHINE LEARNING IS TRANSFORMING HEALTHCARE



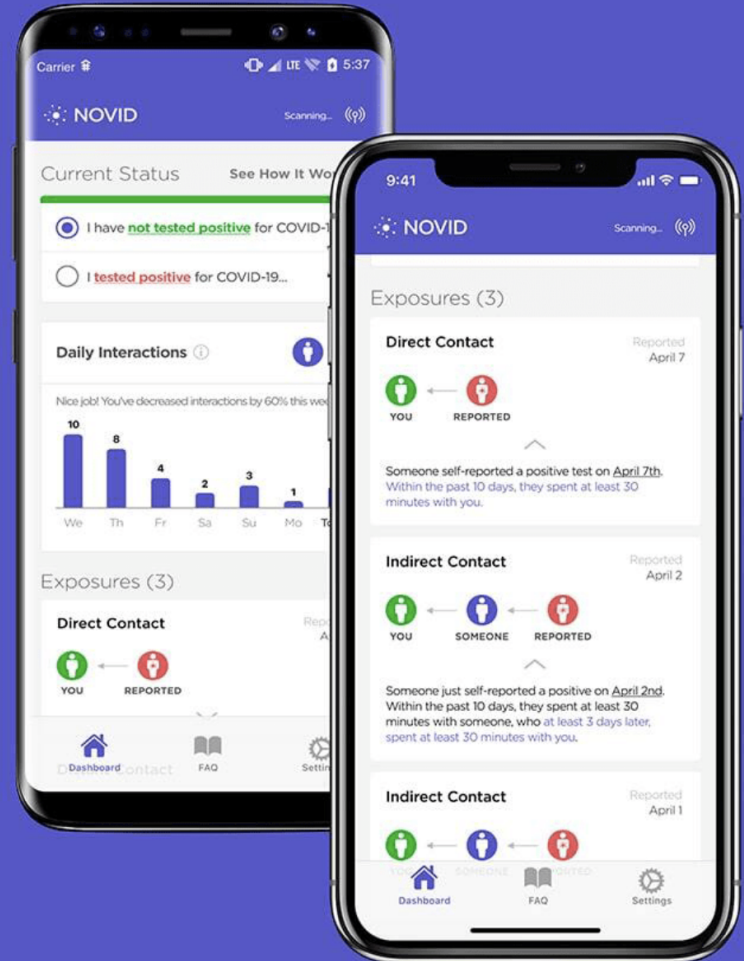
DATA PRIVACY VS UTILITY



NOVID

Stop the Spread.

Carnegie
Mellon
University



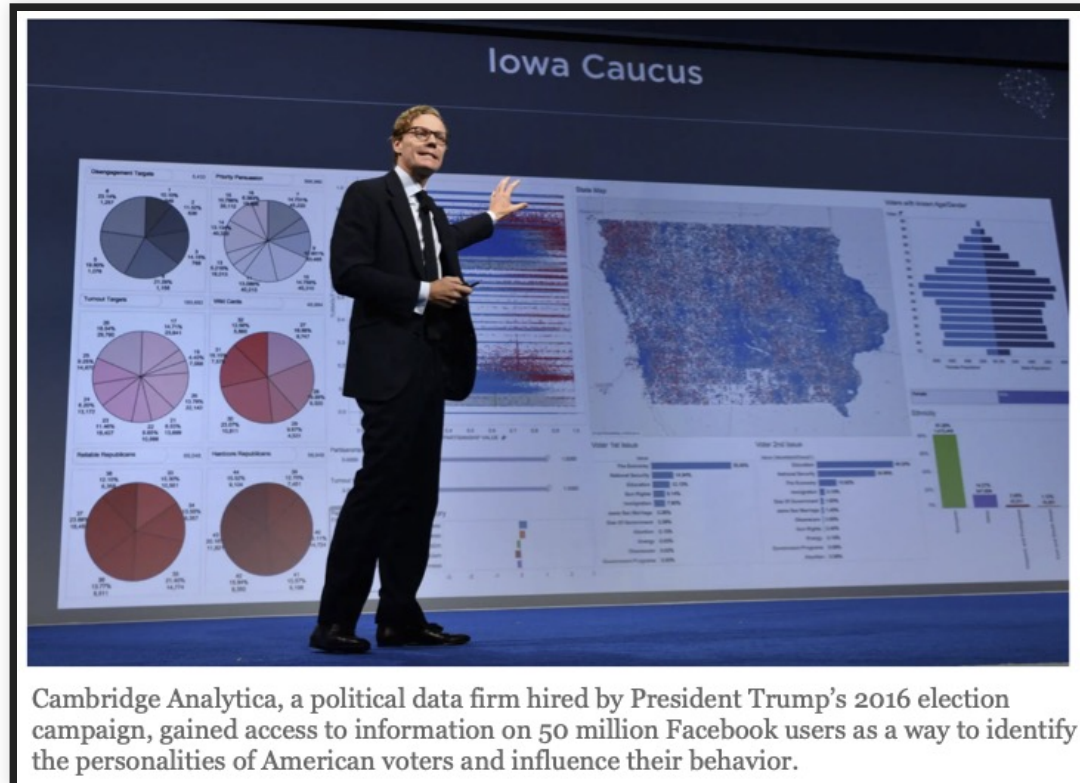
DATA PRIVACY VS UTILITY

Apple Fights Court Order to Unlock San Bernardino Shooter's iPhone

19 février 2016



DATA PRIVACY VS UTILITY



Cambridge Analytica, a political data firm hired by President Trump's 2016 election campaign, gained access to information on 50 million Facebook users as a way to identify the personalities of American voters and influence their behavior.

- ML can leverage data to greatly improve utility for individuals and society
- But unrestrained collection & use of data can enable abuse and harm!
- **Viewpoint:** Users should be given an ability to learn and control how their data is collected and used

BEST PRACTICES FOR ML & DATA PRIVACY


BEST PRACTICES FOR ML & DATA PRIVACY

- Data collection & processing
 - Only collect and store what you need
 - Remove sensitive attributes, anonymize, or aggregate

DATA ANONYMIZATION

ID	Age	Zipcode	Diagnosis
1	28	13053	Heart Disease
2	29	13068	Heart Disease
3	21	13068	Viral Infection
4	23	13053	Viral Infection
5	50	14853	Cancer
6	55	14853	Heart Disease
7	47	14850	Viral Infection
8	49	14850	Viral Infection

k-anonymization



ID	Age	Zipcode	Diagnosis
1	[20-30]	130**	Heart Disease
2	[20-30]	130**	Heart Disease
3	[20-30]	130**	Viral Infection
4	[20-30]	130**	Viral Infection
5	[40-60]	148**	Cancer
6	[40-60]	148**	Heart Disease
7	[40-60]	148**	Viral Infection
8	[40-60]	148**	Viral Infection

- Simply removing explicit identifiers (e.g., name) is often not enough
 - {ZIP, gender, birthdate} can identify 87% of Americans (L. Sweeney)
- k-anonymization: Identity-revealing data tuples appear in at least k rows
 - Suppression: Replace certain values in columns with an asterisk
 - Generalization: Replace individual values with broader categories

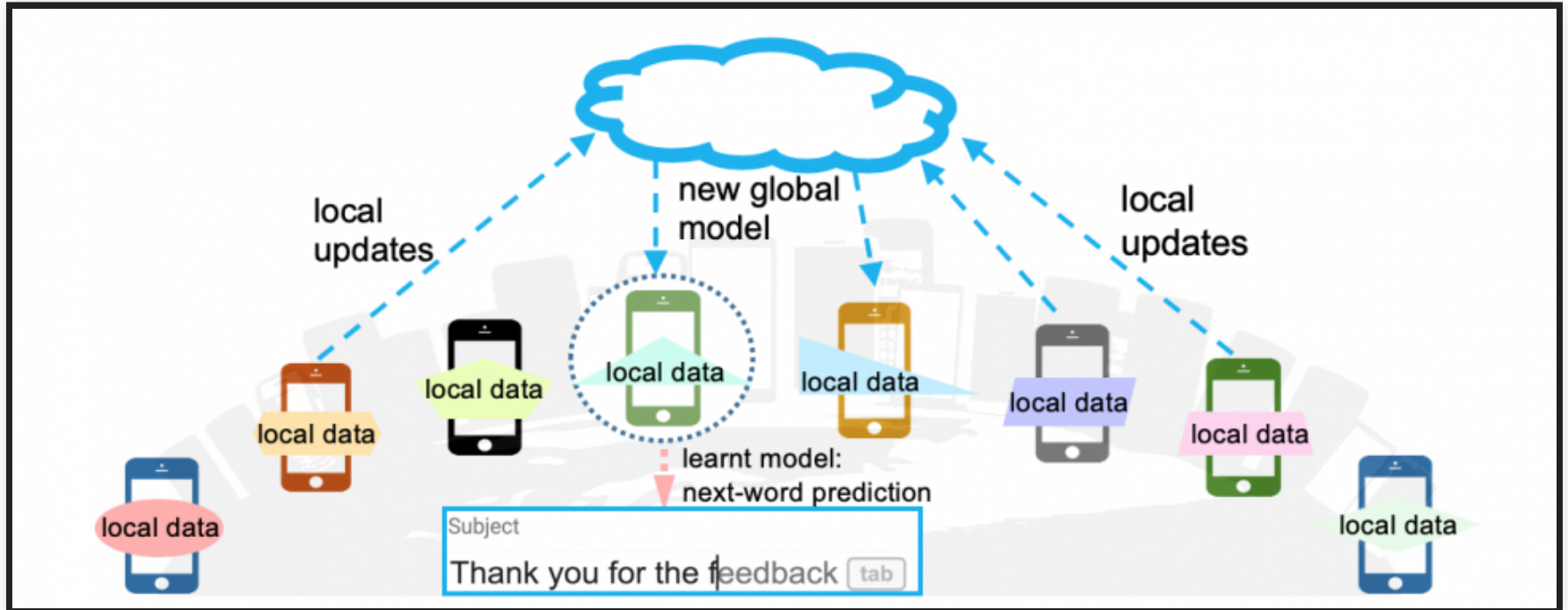
BEST PRACTICES FOR ML & DATA PRIVACY

- Data collection & processing
 - Only collect and store what you need
 - Remove sensitive attributes, anonymize, or aggregate

BEST PRACTICES FOR ML & DATA PRIVACY

- Data collection & processing
 - Only collect and store what you need
 - Remove sensitive attributes, anonymize, or aggregate
- Training: Local, on-device processing if possible
 - Federated learning

FEDERATED LEARNING



- Train a global model using local data stored across multiple edge devices
- Local devices push only model updates, not the raw data
- Improved data privacy & leveraging local processing power; but increased network communication and other security risks (e.g., backdoor injection)

BEST PRACTICES FOR ML & DATA PRIVACY

- Data collection & processing
 - Only collect and store what you need
 - Remove sensitive attributes, anonymize, or aggregate
- Training: Local, on-device processing if possible
 - Federated learning

BEST PRACTICES FOR ML & DATA PRIVACY

- Data collection & processing
 - Only collect and store what you need
 - Remove sensitive attributes, anonymize, or aggregate
- Training: Local, on-device processing if possible
 - Federated learning
- Basic security practices
 - Encryption & authentication
 - Provenance: Track data sources and destinations

BEST PRACTICES FOR ML & DATA PRIVACY

- Data collection & processing
 - Only collect and store what you need
 - Remove sensitive attributes, anonymize, or aggregate
- Training: Local, on-device processing if possible
 - Federated learning
- Basic security practices
 - Encryption & authentication
 - Provenance: Track data sources and destinations
- Provide transparency to users
 - Clearly explain what data is being collected and why

BEST PRACTICES FOR ML & DATA PRIVACY

- Data collection & processing
 - Only collect and store what you need
 - Remove sensitive attributes, anonymize, or aggregate
- Training: Local, on-device processing if possible
 - Federated learning
- Basic security practices
 - Encryption & authentication
 - Provenance: Track data sources and destinations
- Provide transparency to users
 - Clearly explain what data is being collected and why
- Understand and follow the data protection regulations!
 - General Data Protection Regulation (GDPR)
 - California Consumer Privacy Act (CCPA)
 - Domain-specific regulations: HIPPA (healthcare), FERPA (educational)

GENERAL DATA PROTECTION REGULATION (GDPR)

GENERAL DATA PROTECTION REGULATION (GDPR)

- Introduced by the European Union (EU) in 2016

GENERAL DATA PROTECTION REGULATION (GDPR)

- Introduced by the European Union (EU) in 2016
- Organizations must state:
 - What personal data is being collected & stored
 - Purpose(s) for which the data will be used
 - Other entities that the data will be shared with

GENERAL DATA PROTECTION REGULATION (GDPR)

- Introduced by the European Union (EU) in 2016
- Organizations must state:
 - What personal data is being collected & stored
 - Purpose(s) for which the data will be used
 - Other entities that the data will be shared with
- Organizations must receive explicit consent from users

GENERAL DATA PROTECTION REGULATION (GDPR)

- Introduced by the European Union (EU) in 2016
- Organizations must state:
 - What personal data is being collected & stored
 - Purpose(s) for which the data will be used
 - Other entities that the data will be shared with
- Organizations must receive explicit consent from users
- Each user must be provided with the ability to:
 - View, modify and delete any personal data

GENERAL DATA PROTECTION REGULATION (GDPR)

- Introduced by the European Union (EU) in 2016
- Organizations must state:
 - What personal data is being collected & stored
 - Purpose(s) for which the data will be used
 - Other entities that the data will be shared with
- Organizations must receive explicit consent from users
- Each user must be provided with the ability to:
 - View, modify and delete any personal data
- Compliance & enforcement
 - Complaints are filed against non-compliant organizations
 - A failure to comply may result in heavy penalties!

PRIVACY CONSENT AND CONTROL



Your data. Your experience.

TechCrunch is part of [Verizon Media](#). We and [our partners](#) will store and/or access information on your device through the use of cookies and similar technologies, to display personalised ads and content, for ad and content measurement, audience insights and product development.

Your personal data that may be used

- Information about your device and internet connection, including your IP address
- Browsing and search activity while using Verizon Media websites and apps
- [Precise location](#)

Find out more about how we use your information in our [Privacy Policy](#) and [Cookie Policy](#).

To enable Verizon Media and our partners to process your personal data select '**I agree**', or select '**Manage settings**' for more information and to manage your choices. You can change your choices at any time by visiting [Your Privacy Controls](#).

I agree

Manage settings

Amazon hit with \$886m fine for alleged data law breach

🕒 30 July 2021



GETTY IMAGES

BEST PRACTICES FOR ML & DATA PRIVACY

Be ethical and responsible with user data! Think about potential harms to users & society, caused by (mis-)handling of personal data

- Data collection & processing
- Training: Local, on-device processing if possible
- Basic security practices
- Provide transparency to users
- Understand and follow the data protection regulations!

SUMMARY

- Security requirements: Confidentiality, integrity, availability
- Threat modeling to identify security requirements & attacker capabilities
- ML-specific attacks on training data, telemetry, or the model
 - Poisoning attack on training data to influence predictions
 - Evasion attacks to shape input data to achieve intended predictions (adversarial learning)
 - Model inversion attacks for privacy violations
- Security design at the system level
 - Principle of least privilege
 - Isolation & compartmentalization
- AI can be used for defense (e.g. anomaly detection)
- **Key takeaway:** Adopt a security mindset! Assume all components may be vulnerable in one way or another. Design your system to explicitly reduce the impact of potential attacks