Learning-based Network Intrusion Detection: Are We There Yet?

Gregory Blanc
IMT/Télécom SudParis, Institut Polytechnique de Paris

Cyber in Berry 2.0
INSA CVL, Bourges, 2024/07/15

$whoami

- faculty at **Télécom SudParis**, an **IMT** school, member of **IP Paris**
- researcher at **SCN** *(Sécurité et Confiance Numérique)*, a team of the **SAMOVAR** lab
- associated member of **LINCS**
- head of the **SSR** *(Sécurité des Systèmes et Réseaux)* specialization curriculum
- interested in **network security**, **network virtualization**, **machine learning for cybersecurity**
- holds a Ph.D degree from *Nara Institute of Science and Technology* (**NAIST**), Japan
- holds a *Mastère Spécialisé* in *Networks and Information Security* and a *Diplôme d'Ingénieur* from **ESIEA**
- led the **SWAN** *(Security of Web ApplicatioNs)* WG at **WIDE**, Japan
- worked as a security solutions integrator at **BT CyberNetworks**

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. Machine learning-based network intrusion detection

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. Machine learning-based network intrusion detection
   - traffic classification, anomaly detection

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. Machine learning-based network intrusion detection
   - traffic classification, anomaly detection
   - traffic generation (testbed, synthesis, quality)

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. Machine learning-based network intrusion detection
   - traffic classification, anomaly detection
   - traffic generation (testbed, synthesis, quality)
   - intrusion detector evaluation (reproducibility, robustness)

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. Machine learning-based network intrusion detection
   - traffic classification, anomaly detection
   - traffic generation (testbed, synthesis, quality)
   - intrusion detector evaluation (reproducibility, robustness)
   - adversarial examples against intrusion detection systems

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. Machine learning-based network intrusion detection
   - traffic classification, anomaly detection
   - traffic generation (testbed, synthesis, quality)
   - intrusion detector evaluation (reproducibility, robustness)
   - adversarial examples against intrusion detection systems
   - collaborative intrusion detection

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. Machine learning-based network intrusion detection
   - traffic classification, anomaly detection
   - traffic generation (testbed, synthesis, quality)
   - intrusion detector evaluation (reproducibility, robustness)
   - adversarial examples against intrusion detection systems
   - collaborative intrusion detection
2. Network attack mitigation using programmable networks
   - adverse impact-sensitive, automated selection of remediations
   - automated deployment of security policy/measures

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. Machine learning-based network intrusion detection
   - traffic classification, anomaly detection
   - traffic generation (testbed, synthesis, quality)
   - intrusion detector evaluation (reproducibility, robustness)
   - adversarial examples against intrusion detection systems
   - collaborative intrusion detection
2. Network attack mitigation using programmable networks
   - adverse impact-sensitive, automated selection of remediations
   - automated deployment of security policy/measures
     - resource-security tradeoff optimization (MDP, algebraic constraint solving)
     - reprogrammable network security functions deployment
     - monitoring-based network policy verification

Research Interests

keywords: *network security, network programming, interactions between AI and cybersecurity*

1. **Machine learning-based network intrusion detection**
   - traffic classification, anomaly detection
   - traffic generation (testbed, synthesis, quality)
   - intrusion detector evaluation (reproducibility, robustness)
   - adversarial examples against intrusion detection systems
   - collaborative intrusion detection
2. Network attack mitigation using programmable networks
   - adverse impact-sensitive, automated selection of remediations
   - automated deployment of security policy/measures

Team and Projects

Contributors

- Mustafizur R. Shahid (Ph.D, 2017–2021)
- Houda Jmila (Postdoc, 2018–2023)
- Marwan Lazrag (Engineer, 2019–)
- Paul Peseux (Intern, 2019)
- PH Mignot (Engineer, 2021–2022)
- Adrien Schoen (Ph.D, 2021–)
- Solayman Ayoubi (Ph.D, 2022–)
- Sara Chennoufi (Ph.D, 2022–)
- Marin Stamm (Master, 2022–2023)
- Satoshi Okada (Ph.D, 2023)
- Matthieu Mouzaoui (Ph.D, 2024–)

Projects and Fundings

- *Futur & Ruptures Ph.D Grant* (IMT, 2017–2021)
- *CEF VARIoT* (2019–2022)
- *H2020 SPARTA – CAPE* (2019–2022)
- *France Relance Beyond5G* (2021–2024)
- *ANR GRIFIN* (2021–2025)
- *CIEDS CERES* (2021–2025)
- *PEPR Cyber: SuperviZ* (2022–2028)

Collaborators: SAMOVAR/SCN (H. Debar, C. Kiennert), IRISA/Pirat (PF Gimenez, L. Mé, Y. Han, F. Majorczyk), NICT (F. Charmet, T. Takahashi, H.C. Tanuwidjaja), T. Silverston (LORIA), S. Tixeuil (LIP6), Z. Zhang (IMT Nord Europe)

Outline

Global Shortage of Cybersecurity Experts

The cybersecurity industry has an urgent talent shortage. Here's how to plug the gap

World Economic Forum, April 2024

Global Shortage of Cybersecurity Experts

The cybersecurity industry has an urgent talent shortage. Here's how to plug the gap

World Economic Forum, April 2024

La pénurie de talents en cybersécurité, une épine dans le pied des entreprises françaises

L'Usine Digitale, July 2024

Global Shortage of Cybersecurity Experts

The cybersecurity industry has an urgent talent shortage. Here's how to plug the gap

World Economic Forum, April 2024

La pénurie de talents en cybersécurité, une épine dans le pied des entreprises françaises

L'Usine Digitale, July 2024

Fatigue and shortages: cyber teams intentionally underreporting breaches

Cybernews, May 2024

Ever-evolving Threat Landscape

Cloud
- migration from on-premise to remote services
- lack of network control and observability

Ever-evolving Threat Landscape

Cloud
- migration from on-premise to remote services
- lack of network control and observability

5G and IoT
- 5G enables customized IoT network slices
- IoT devices often vulnerable and, now exposed

Ever-evolving Threat Landscape

Cloud
- migration from on-premise to remote services
- lack of network control and observability

5G and IoT
- 5G enables customized IoT network slices
- IoT devices often vulnerable and, now exposed

ICS
- more remote access to Industrial Control Systems
- critical ICS rely on obsolete network protocols

Ever-evolving Threat Landscape

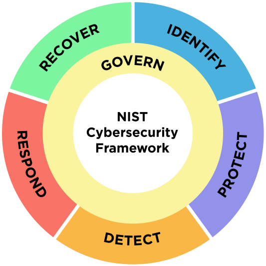
Cloud
- migration from on-premise to remote services
- lack of network control and observability

5G and IoT
- 5G enables customized IoT network slices
- IoT devices often vulnerable and, now exposed

ICS
- more remote access to Industrial Control Systems
- critical ICS rely on obsolete network protocols

(Gen)AI
- with the advent of LLMs, GenAI tools are pervasive
- AI risks are emerging and not well understood

Opportunities to use AI for Cybersecurity



NIST Cybersecurity Framework, February 2024

- Alleviate experts' load
- Automate complex tasks
- Analyse vast amount of data
- Uncover underlying patterns
- Support decision making
- Anticipate future threats

Outline

What is an intrusion?

NIST definition (2007)

What is an intrusion?

NIST definition (2007)

(or *incident*)

A violation or imminent threat of violation of computer security policies, acceptable use policies, or standard security practices.

What is an intrusion?

NIST definition (2007)

(or *incident*)

A violation or imminent threat of violation of computer security policies, acceptable use policies, or standard security practices.

Additionally,

What is an intrusion?

NIST definition (2007)

(or *incident*)

A violation or imminent threat of violation of computer security policies, acceptable use policies, or standard security practices.

Additionally,

Incidents have many causes, such as malware, attackers gaining unauthorized access to systems from the Internet, and authorized users of systems who misuse their privileges or attempt to gain additional privileges for which they are not authorized.

What is an intrusion?

NIST definition (2007)

(or *incident*)

A violation or imminent threat of violation of computer security policies, acceptable use policies, or standard security practices.

Additionally,

Incidents have many causes, such as malware, attackers gaining unauthorized access to systems from the Internet, and authorized users of systems who misuse their privileges or attempt to gain additional privileges for which they are not authorized.

ANSSI definition (CyberDico, 2024)

Intrusion is the act of a person or object entering a defined space (physical, logical, relational) where **its presence is not desired**.

What is an intrusion?

NIST definition (2007)

(or *incident*)

A violation or imminent threat of violation of computer security policies, acceptable use policies, or standard security practices.

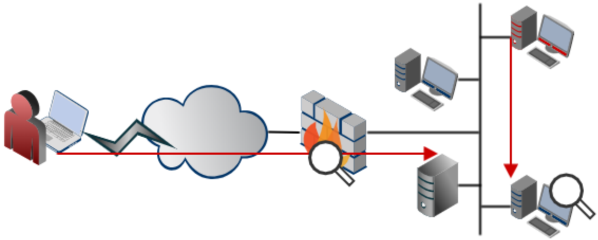
Additionally,

Incidents have many causes, such as malware, attackers gaining unauthorized access to systems from the Internet, and authorized users of systems who misuse their privileges or attempt to gain additional privileges for which they are not authorized.

ANSSI definition (CyberDico, 2024)

Intrusion is the act of a person or object entering a defined space (physical, logical, relational) where **its presence is not desired**.

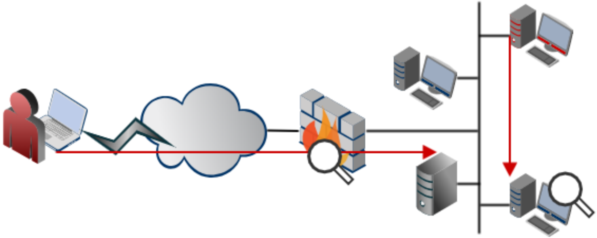
Intrusion Detection



Alert on any **suspicious** activity enabling later filtering or correlation

- What is suspicious?

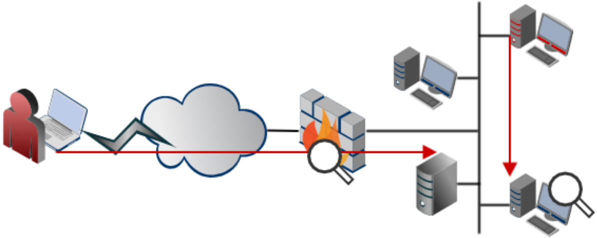
Intrusion Detection



Alert on any **suspicious** activity enabling later filtering or correlation

- What is suspicious?
  - **misuse**: *activity known to be malicious*

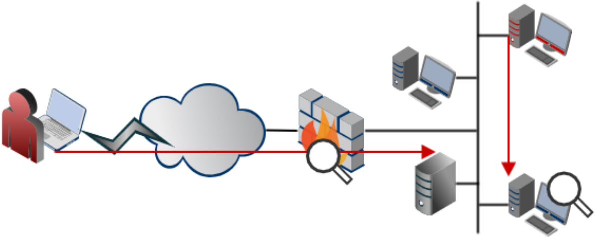
Intrusion Detection



Alert on any **suspicious** activity enabling later filtering or correlation

- What is suspicious?
  - **misuse**: *activity known to be malicious*
  - **anomaly**: *activity deviant from normal*

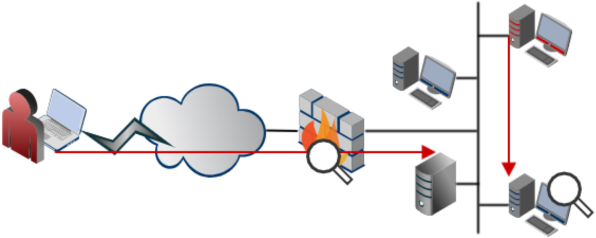
Intrusion Detection



Alert on any **suspicious** activity enabling later filtering or correlation

- What is suspicious?
  - **misuse**: *activity known to be malicious*
  - **anomaly**: *activity deviant from normal*
- How to capture suspicious activities?

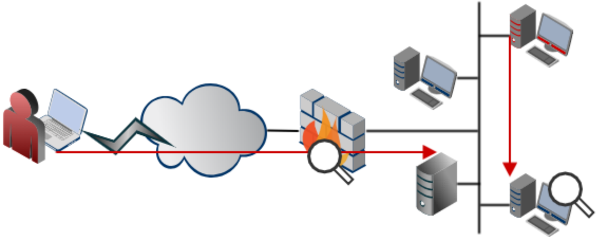
Intrusion Detection



Alert on any **suspicious** activity enabling later filtering or correlation

- What is suspicious?
  - **misuse**: *activity known to be malicious*
  - **anomaly**: *activity deviant from normal*
- How to capture suspicious activities?
  - at the host: process, log, file, etc.

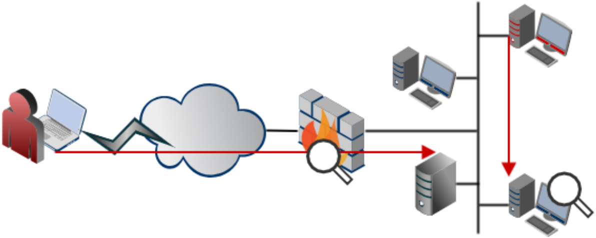
Intrusion Detection



Alert on any **suspicious** activity enabling later filtering or correlation

- What is suspicious?
  - **misuse**: *activity known to be malicious*
  - **anomaly**: *activity deviant from normal*
- How to capture suspicious activities?
  - at the host: process, log, file, etc.
  - in the network: flow, packet headers, payloads, etc.

Intrusion Detection



Alert on any **suspicious** activity enabling later filtering or correlation

- What is suspicious?
  - **misuse**: *activity known to be malicious*
  - **anomaly**: *activity deviant from normal*
- How to capture suspicious activities?
  - at the host: process, log, file, etc.
  - in the network: flow, packet headers, payloads, etc.

Huge volume of activities incur *longer* processing time

Misuse detection

Approach *mostly* attack signatures

Features packet headers, flow stats, TCP connections, etc.

Trends data mining and machine learning on labeled traffic datasets

Challenges
- lack of datasets (existence, diversity, freshness, reliability)
- frequency of model re-training

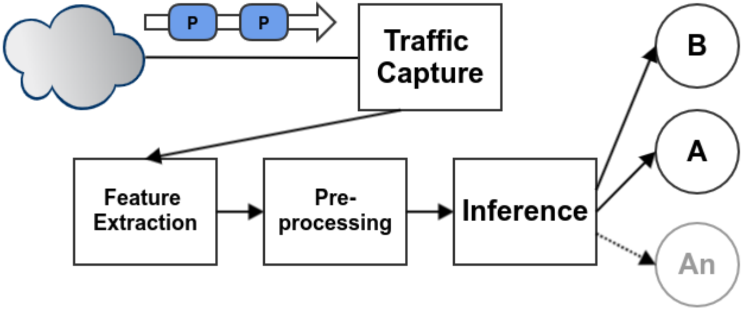
Anomaly detection

Approach	(normal) behavioural profiles
Learning	unsupervised, semi-supervised, supervised
Challenges	■ cleanliness of datasets ■ accuracy of normal behaviour ■ high false positive rate

Works well with *low-entropy* normal behaviour

Outline

Detection's ML Pipeline



Inference refers to the **trained** detection model decision-making

Intrusion Detection as a Classification Task

Misuse detection

Anomaly detection

Intrusion Detection as a Classification Task

Misuse detection

- Each class encodes a pattern of features, akin to a **signature**
- Model is **limited** to attack classes in the training set
- Alleviates nonetheless the **pain and risk** of manual signature design

Anomaly detection

Intrusion Detection as a Classification Task

Misuse detection

- Each class encodes a pattern of features, akin to a **signature**
- Model is **limited** to attack classes in the training set
- Alleviates nonetheless the **pain and risk** of manual signature design

Anomaly detection

- Training on **benign data only** yields patterns of normal behaviour
- The trained detection model enables detecting **deviations**
- Lacks precision as an anomaly **does not indicate** malice

Intrusion Detection as a Classification Task

Misuse detection

- Each class encodes a pattern of features, akin to a **signature**
- Model is **limited** to attack classes in the training set
- Alleviates nonetheless the **pain and risk** of manual signature design

Anomaly detection

- Training on **benign data only** yields patterns of normal behaviour
- The trained detection model enables detecting **deviations**
- Lacks precision as an anomaly **does not indicate** malice

- Training classifiers on **huge** amounts of data still seems profitable

Intrusion Detection as a Classification Task

Misuse detection

- Each class encodes a pattern of features, akin to a **signature**
- Model is **limited** to attack classes in the training set
- Alleviates nonetheless the **pain and risk** of manual signature design

Anomaly detection

- Training on **benign data only** yields patterns of normal behaviour
- The trained detection model enables detecting **deviations**
- Lacks precision as an anomaly **does not indicate** malice

- Training classifiers on **huge** amounts of data still seems profitable
- Performance depends on the data **quality**, i.e., *representation*, *representativeness*, etc.

Intrusion Detection as a Classification Task

Misuse detection

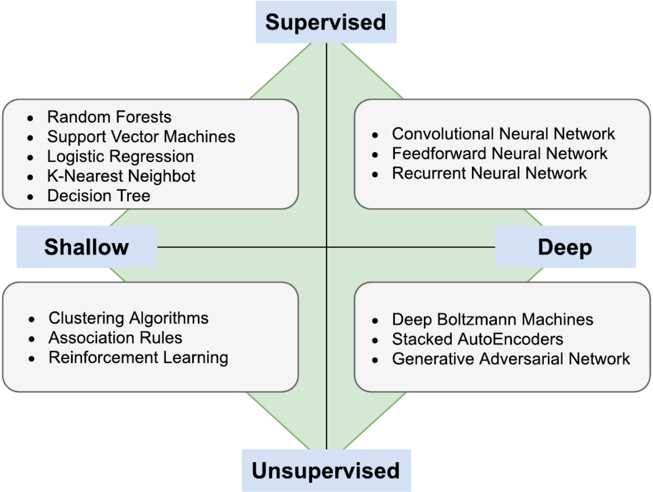
- Each class encodes a pattern of features, akin to a **signature**
- Model is **limited** to attack classes in the training set
- Alleviates nonetheless the **pain and risk** of manual signature design

Anomaly detection

- Training on **benign data only** yields patterns of normal behaviour
- The trained detection model enables detecting **deviations**
- Lacks precision as an anomaly **does not indicate** malice

- Training classifiers on **huge** amounts of data still seems profitable
- Performance depends on the data **quality**, i.e., *representation*, *representativeness*, etc.
- Myth: *contrary to signatures, anomaly-based detection uses ML* [1]

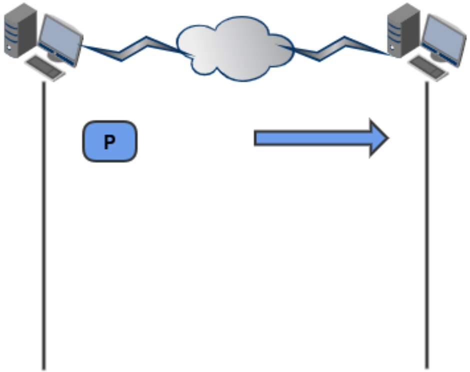
Most Used ML Algorithms for IDS [1]



How to Capture Network Traffic?

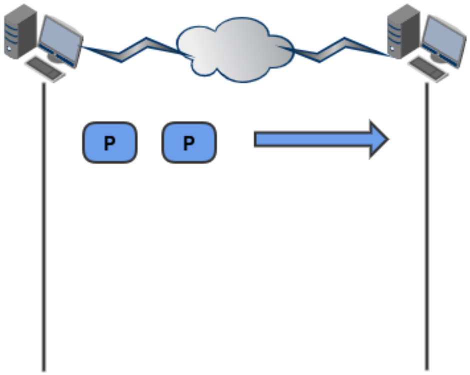
Network traffic is the set of **communications** exchanged in a network from a vantage point

How to Capture Network Traffic?



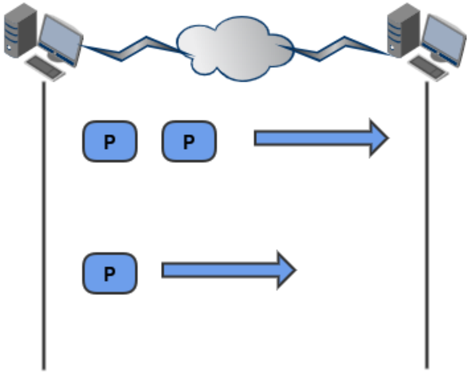
Between two hosts, we can observe **packet by packet**

How to Capture Network Traffic?



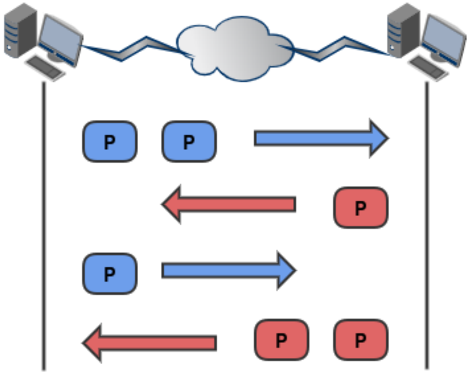
Between two hosts, we can observe **a sequence of packets**

How to Capture Network Traffic?



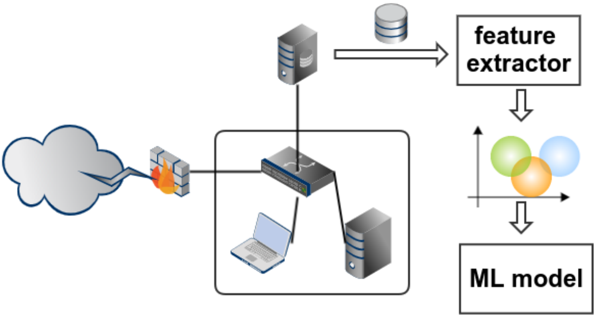
A **flow** is defined as a sequence of packet **sharing common characteristics**

How to Capture Network Traffic?



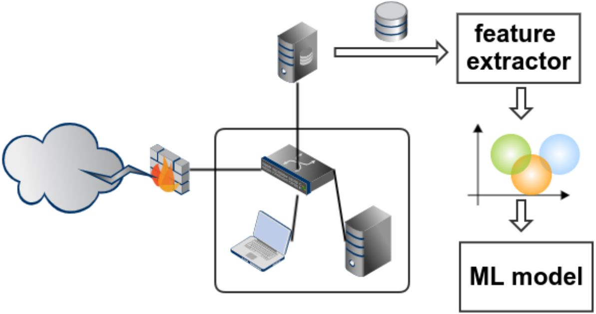
A **bidirectional flow** considers both directions

How to Represent Network Traffic?



1. Traffic is captured from the data plane as `pcap`

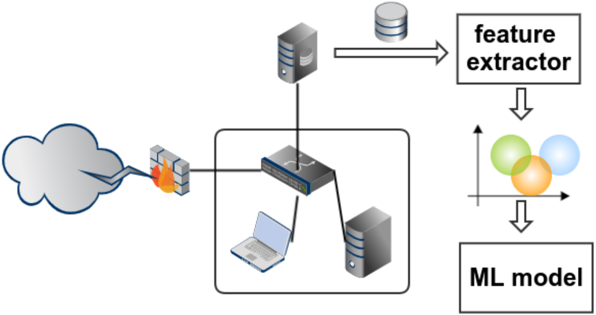
How to Represent Network Traffic?



2. A **feature extractor** extracts information from the `pcap` to represent the traffic in a **feature space**

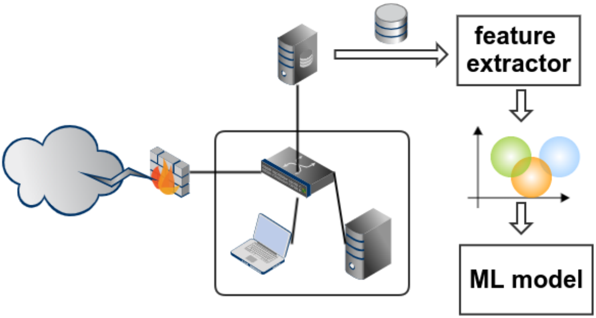
TELECOM SudParis
IP PARIS

How to Represent Network Traffic?



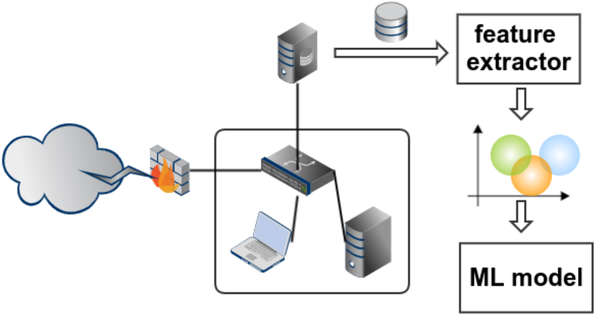
2.a. **Packet-level** information deals with the flow identifier (at least, `src IP`, `src Port`, `dst IP`, `dst Port`, `L4 Protocol`) and related header information

How to Represent Network Traffic?



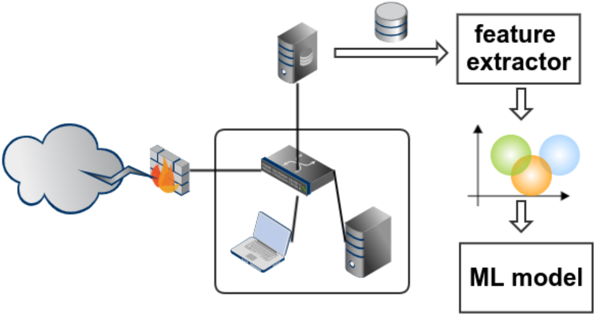
2.b. **Packet payload** may also be represented but often <u>absent</u> (*due to privacy or encryption*)

How to Represent Network Traffic?



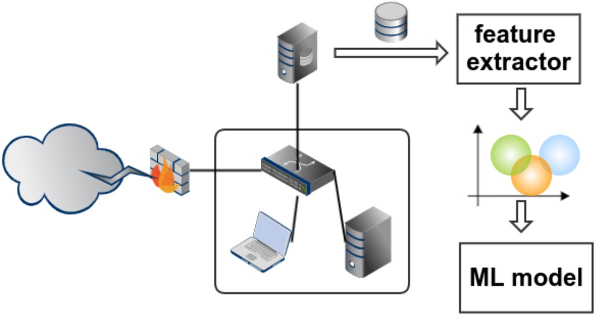
2.c. **Flow-level** information attempts at **summarizing** a sequence of packets sharing the same flow identifier (length, duration, IAT, etc.)

How to Represent Network Traffic?



3. Among other preprocessing steps, the dimension of the feature space can be reduced through **feature selection** (*manual*) or **dimension reduction**

How to Represent Network Traffic?



X. Alternatively, some approaches may resort to **feature learning**, which automatically discovers an appropriate **representation**

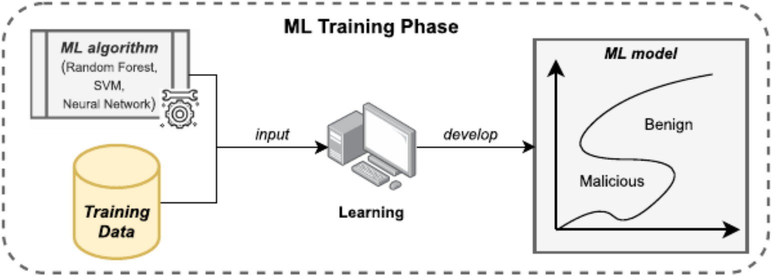
Flow Information

Flow-level datasets are very popular to briefly represent network traffic. Here is a NetFlow [2] based feature set [3].

Feature	Description		
IPv4_Src_Addr	–	L7_Proto	–
IPv4_Dst_Addr	–	In_Bytes	Incoming number of bytes
L4_Src_Port	–	Out_Bytes	Outgoing number of bytes
L4_Dst_Port	–	In_Pkts	Incoming number of packets
Protocol	IP protocol identifier	Out_Pkts	Outgoing number of packets
TCP_Flags	Cumulative of all TCP flags	Flow_Duration	Flow duration in milliseconds

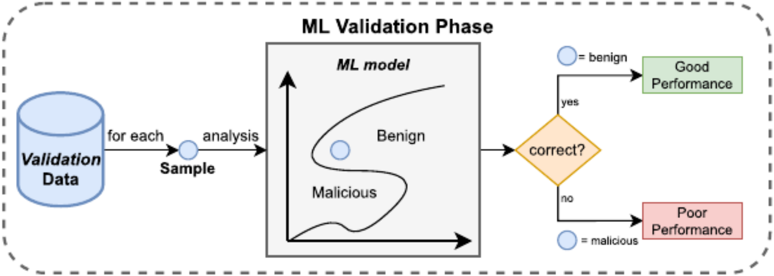
Other wider feature sets of dimensions 43 [4] and 83 [5] using NetFlow and CICFlow formats, respectively.

How to Evaluate an ML-based NIDS?



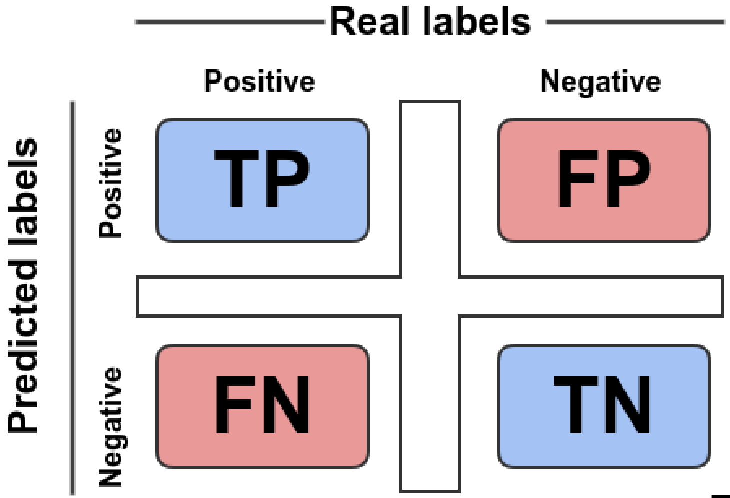
Pictures from Apruzzese et al. [1]

How to Evaluate an ML-based NIDS?



Pictures from Apruzzese et al. [1]

How to Evaluate an ML-based NIDS?



Classification Metrics [6]

Evaluating an IDS is often considered a binary classification problem. Leveraging the confusion matrix, we can measure:

- **Accuracy**: $\frac{TN+TP}{TP+FP+TN+FN)}$ (overall success rate)
- **Precision**: $\frac{TP}{TP+FP}$ (aka *positive predicted value*)
- **Detection Rate**: $\frac{TP}{TP+FN}$ (aka *sensitivity* or *recall*)
- **True Negative Rate**: $\frac{TN}{TN+FP}$ (aka *specificity*)
- **False Positive Rate**: $\frac{FP}{FP+TN} = 1 - TNR$ (aka *fall-out*)
- **F-measure**: $2 \times \frac{precision \times recall}{precision + recall}$
- **Receiver Operating Characteristic curve**: plot of the *sensitivity* as a function of $1-$ *specificity*

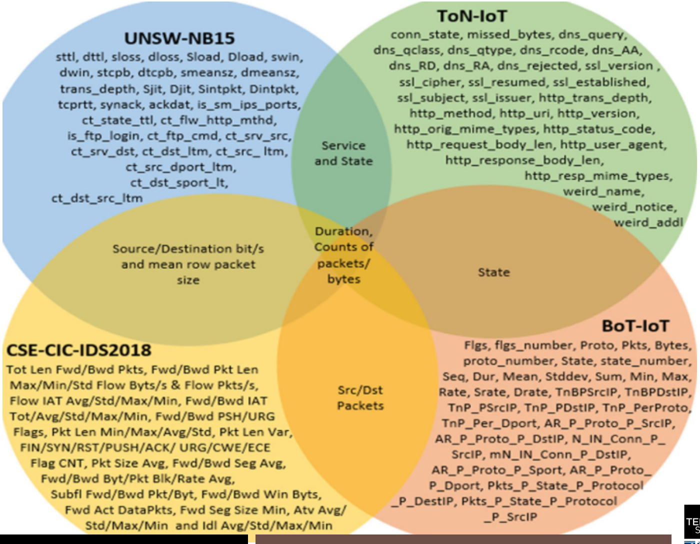
Datasets

- Packet-based: available in pcap, contains payload, metadata depending on used protocols
- Flow-based: condensed metadata-rich information, no payload, aggregates all packet sharing some properties (e.g., 5-tuple) within a time window
- Other data: hybrid data set (packet/flow, network/host)

Ring et al. [7] surveyed existing datasets and grouped them:

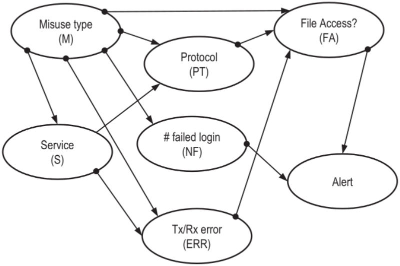
- public? attacks?
- metadata?
- which format
- the volume of data and its duration
- the kind of traffic and the type of network
- balanced? labeled? predefined splits?

Towards a Standard Feature Set [4]



Some Sample Shallow Detection Models [8]

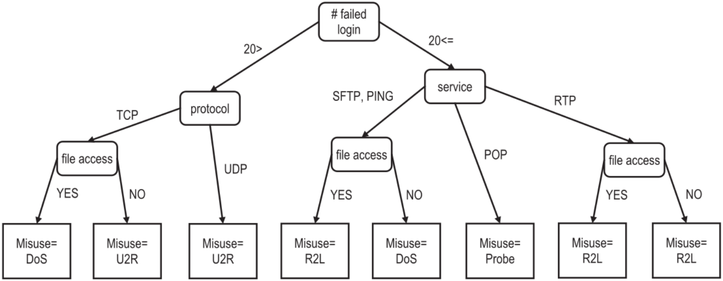
- Bayesian Network



File Access state input variables and values	P(FA = True)	P(FA = False)
M=R2H, PT=NSF, ERR=0	0.95	0.05
M=R2H, PT=FTP, ERR=0	0.99	0.01
M=Probe, PT=none, ERR=50%	0.80	0.20
M=Probe, PT=PING, ERR=0	0.50	0.50
M=DoS, PT=POP, ERR=100%	0.80	0.20
M= DoS, PT=HTTP, ERR=50%	0.90	0.10

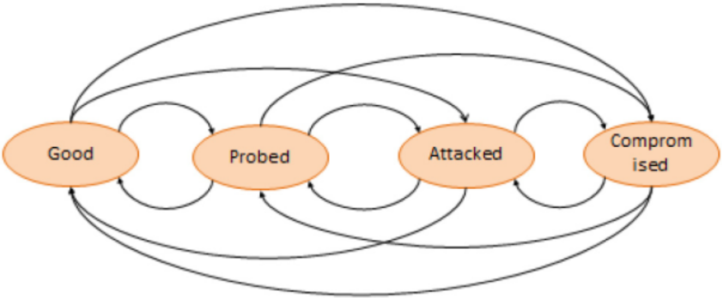
Some Sample Shallow Detection Models [8]

- Decision Tree



Some Sample Shallow Detection Models [8]

- Hidden Markov Model



Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
    - especially in unsupervised mode (no labeling required)

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]
  - no consistent evaluation methodology

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]
  - no consistent evaluation methodology
  - no consistent performance
    - highly dependent on the type of attack and number of classes
    - scarce number of malicious samples

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]
  - no consistent evaluation methodology
  - no consistent performance
- Current obstacles hamper DL-based IDS research [10]

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]
  - no consistent evaluation methodology
  - no consistent performance
- Current obstacles hamper DL-based IDS research [10]
  - Limited availability of public IDS datasets (small, old, internally generated, private)

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]
  - no consistent evaluation methodology
  - no consistent performance
- Current obstacles hamper DL-based IDS research [10]
  - Limited availability of public IDS datasets (small, old, internally generated, private)
  - Inability to test in operational scenarios (detection rate, speed, memory usage, etc.)

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]
  - no consistent evaluation methodology
  - no consistent performance
- Current obstacles hamper DL-based IDS research [10]
  - Limited availability of public IDS datasets (small, old, internally generated, private)
  - Inability to test in operational scenarios (detection rate, speed, memory usage, etc.)
  - Difficulty to explain decisions (blackbox)

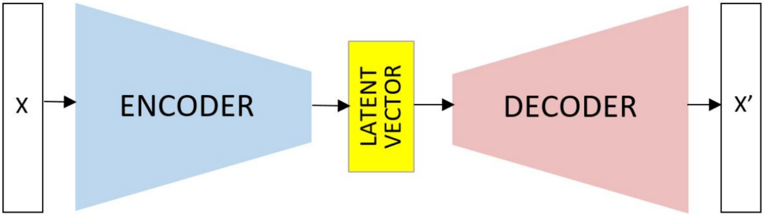
Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]
  - no consistent evaluation methodology
  - no consistent performance
- Current obstacles hamper DL-based IDS research [10]
  - Limited availability of public IDS datasets (small, old, internally generated, private)
  - Inability to test in operational scenarios (detection rate, speed, memory usage, etc.)
  - Difficulty to explain decisions (blackbox)
  - Often tailored to specific threats (vulnerability to concept drift)
    - yet more performant than general detectors [11]

Deep Learning based Intrusion Detection

- ML has been proven successful for intrusion detection [1]
- DL offers opportunities
  - when the rate of new attacks outpace the ability to write and deploy signatures
  - when there is a huge amount (number of samples) of complex data (number of features)
- but DL has not proven to outperform shallow ML [9, 10]
  - no consistent evaluation methodology
  - no consistent performance
- Current obstacles hamper DL-based IDS research [10]
  - Limited availability of public IDS datasets (small, old, internally generated, private)
  - Inability to test in operational scenarios (detection rate, speed, memory usage, etc.)
  - Difficulty to explain decisions (blackbox)
  - Often tailored to specific threats (vulnerability to concept drift)
  - Potential vulnerability to *smart attackers* (e.g., adversarial examples)

Autoencoders (AE)

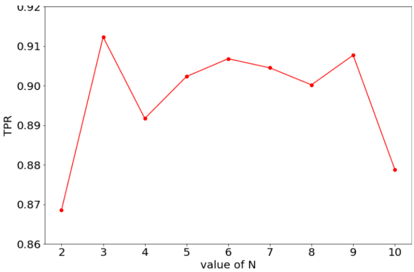


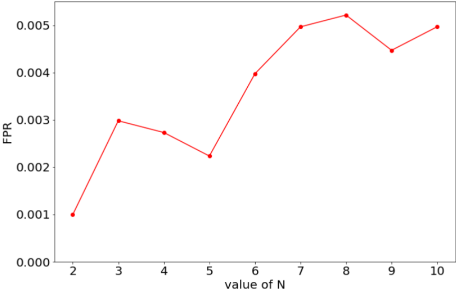
AEs are unsupervised NNs that learn to copy their inputs to their outputs under some constraints [12].

Semi-supervised IoT Anomaly-based IDS [12]

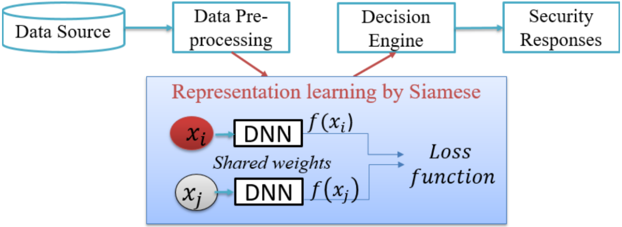
Features
Mean, Median, Min, Max, Standard deviation and Count of the size of the first N packets sent
Mean, Median, Min, Max, Standard deviation and Count of the size of the first N packets received
Mean and Standard deviation of the IAT between the first N packets sent
Mean and Standard deviation of the IAT between the first N packets received

	Bidirectional flows
D-Link Motion Sensor	1074
Nest Security Camera	1055
TP-Link Smart Bulb	1040
TP-Link Smart Plug	858
Total	**4027**





Siamese Network based Feature Learning [13]

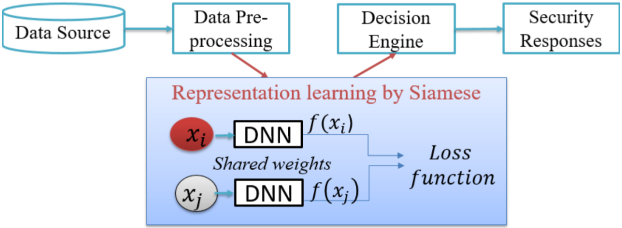


Goal: Minimize $\mathcal{L} = \sum_{i=1}^{N} Loss(D_i, Y_i)$

Cost:

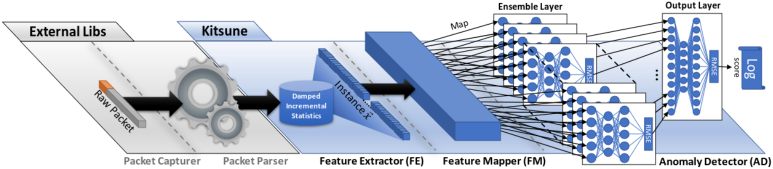
- $\binom{|c_i|}{2}$ for **positive pairs**, i.e., samples from the same class c_i
- $|c_i| \times |c_j|$ for **negative pairs**, i.e., samples from different classes, c_i and c_j

Siamese Network based Feature Learning [13]



It improved binary- and multi-classification results in both **unbalanced** and **small datasets**

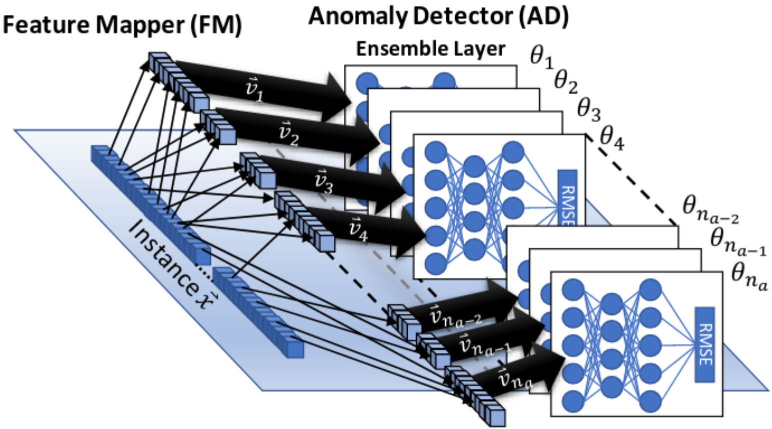
Practical Case Study: Kitsune [14]



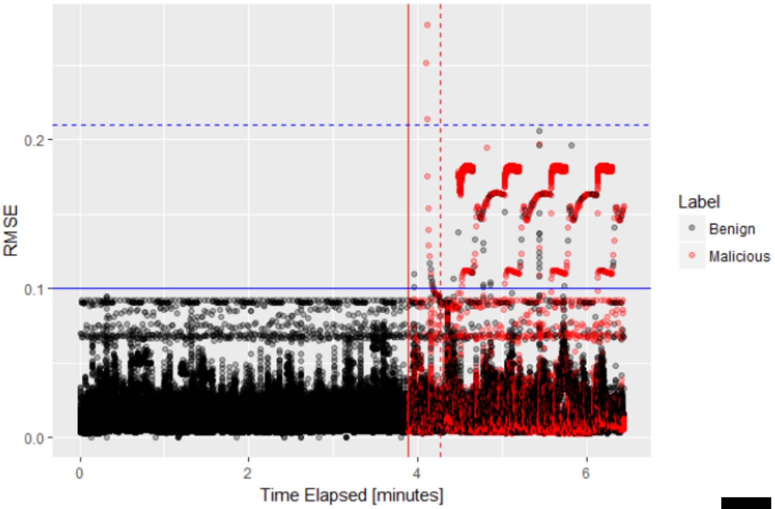
Kitsune is made of 3 main components:

- **Feature Extractor**: to create *n*-feature vectors ($\vec{x}$) that describe packets and the channel they came from
- **Feature Mapper**: to create smaller instances v from $\vec{x}$ according to a learnt mapping
- **Anomaly Detector** (aka *KitNET*): to detect abnormal packet representations v

Practical Case Study: Kitsune [14]



Practical Case Study: Kitsune [14]



ML/DL-based IDS: Takeaways

- IDS is a classification task (either binary or multiclass)

ML/DL-based IDS: Takeaways

- IDS is a classification task (either binary or multiclass)
- Network traffic is represented either at packet-level or flow-level

ML/DL-based IDS: Takeaways

- IDS is a classification task (either binary or multiclass)
- Network traffic is represented either at packet-level or flow-level
- Yet no standardized representation exists (each dataset has its own feature set)

ML/DL-based IDS: Takeaways

- IDS is a classification task (either binary or multiclass)
- Network traffic is represented either at packet-level or flow-level
- Yet no standardized representation exists (each dataset has its own feature set)
- Many ML and DL algorithms have been trialed, with no superiority of the latter on the former

ML/DL-based IDS: Takeaways

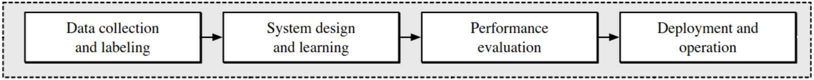
- IDS is a classification task (either binary or multiclass)
- Network traffic is represented either at packet-level or flow-level
- Yet no standardized representation exists (each dataset has its own feature set)
- Many ML and DL algorithms have been trialed, with no superiority of the latter on the former
- Unsupervised approaches are more realistic and may yield better (yet less interpretable) representations

ML/DL-based IDS: Takeaways

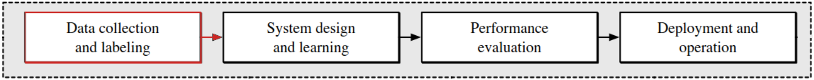
- IDS is a classification task (either binary or multiclass)
- Network traffic is represented either at packet-level or flow-level
- Yet no standardized representation exists (each dataset has its own feature set)
- Many ML and DL algorithms have been trialed, with no superiority of the latter on the former
- Unsupervised approaches are more realistic and may yield better (yet less interpretable) representations
- Anomaly detection is best applied to detect specific behaviours

Outline

Common Pitfalls [15]

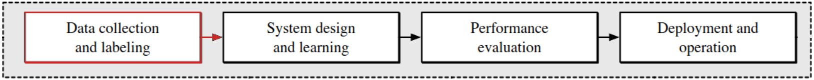


Common Pitfalls [15]



- Ⓐ Sampling bias
- Ⓑ Label inaccuracy

Common Pitfalls [15]

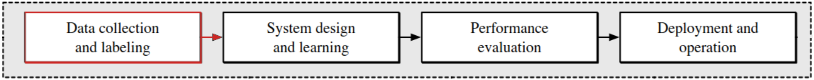


Ⓐ Sampling bias

- collected data does not sufficiently represent the true data distribution of the underlying security problem

Ⓑ Label inaccuracy

Common Pitfalls [15]

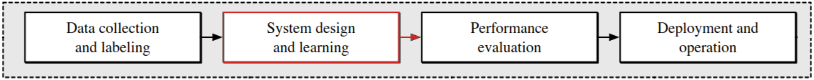


Ⓐ Sampling bias

Ⓑ Label inaccuracy

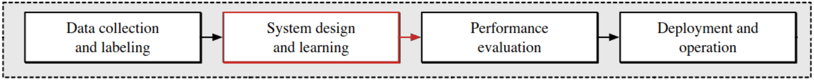
- labels may suffer from changes in their distribution over time
- labels should be verified manually whenever possible

Common Pitfalls [15]



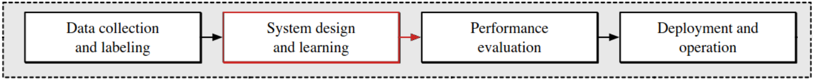
- Ⓒ Data snooping
- Ⓓ Spurrious correlations
- Ⓔ Biased parameters

Common Pitfalls [15]



- (C) Data snooping
  - clumsy data splitting yielding information that should not be available at training time
- (D) Spurrious correlations
- (E) Biased parameters

Common Pitfalls [15]



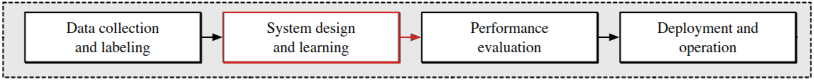
(C) Data snooping

(D) Spurrious correlations

- artifacts that correlate with the task to solve without being related to it
- need to apply explanation techniques

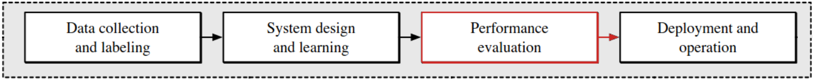
(E) Biased parameters

Common Pitfalls [15]



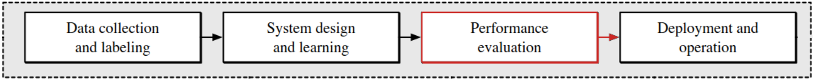
- (C) Data snooping
- (D) Spurrious correlations
- (E) Biased parameters
  - parameters indirectly depending on the test set

Common Pitfalls [15]



- (F) Inappropriate baselines
- (G) Inappropriate measures
- (H) Base rate fallacy [16]

Common Pitfalls [15]



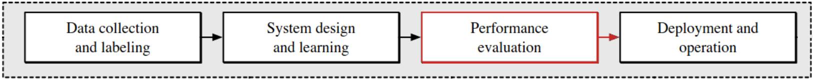
Ⓕ Inappropriate baselines

- need for a simple baseline to motivate the need for a complex ML system

Ⓖ Inappropriate measures

Ⓗ Base rate fallacy [16]

Common Pitfalls [15]



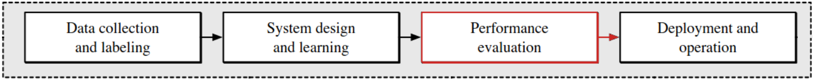
F Inappropriate baselines

G Inappropriate measures

- evaluation should take into account the data specificities

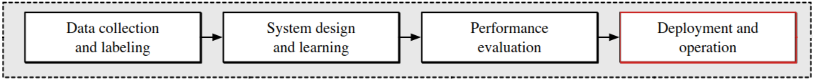
H Base rate fallacy [16]

Common Pitfalls [15]



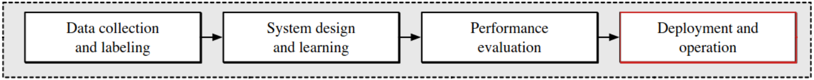
- (F) Inappropriate baselines
- (G) Inappropriate measures
- (H) Base rate fallacy [16]
  - ignoring class imbalance leads to performance overestimation

Common Pitfalls [15]



- (I) Lab-only evaluation
- (J) Inappropriate threat model

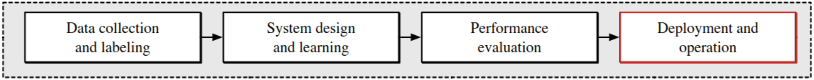
Common Pitfalls [15]



(I) Lab-only evaluation
- detection methods evaluated in a *closed world* setting [17]
- e.g., need to consider temporal and spatial relation in the data [18]

(J) Inappropriate threat model

Common Pitfalls [15]



(I) Lab-only evaluation

(J) Inappropriate threat model

- security of the detection model (*adaptive adversary* [19]) is not considered
- systematically investigate possible vulnerabilities, focusing on white-box attacks

Practical Case Study: Kitsune [14]

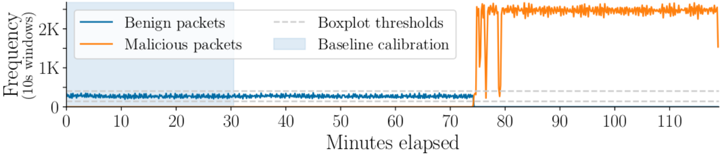
Kitsune's paper has been shown [15] to suffer from:

- Lab-only evaluation (I): a Mirai dataset exhibits crushing attack traffic leading to potential *spurrious correlations* (D)

Practical Case Study: Kitsune [14]

Kitsune's paper has been shown [15] to suffer from:

- Lab-only evaluation (I): a Mirai dataset exhibits crushing attack traffic leading to potential *spurrious correlations* (D)



Practical Case Study: Kitsune [14]

Kitsune's paper has been shown [15] to suffer from:

- Lab-only evaluation (I): a Mirai dataset exhibits crushing attack traffic leading to potential *spurrious correlations* (D)
- Inappropriate baseline (F): an experiment using a *simple boxplot* approach has been shown to exhibit similar AUC, but outperforms Kitsune on FPR

Detector	**AUC**	**TPR**
Kitsune	0.968	0.882
Boxplot	0.998	0.996

Outline

Issues in Testing IDS

Back in 2003, NIST identified several challenges [20]:

- difficulties in collecting attack scripts and victim software
- differing requirements for testing signature based vs. anomaly based IDS
- differing requirements for testing network based vs. host based IDS
- approaches to using background traffic in IDS tests:
  - no background traffic/logs
  - real traffic/logs
  - sanitized traffic/logs
  - generating traffic on a testbed network

Evaluation Metrics

In 2015, IDS evaluation best practices measure (w.r.t. *attack detection*) [21]:

- **Attack detection accuracy**: *accuracy* of an IDS in the presence of *mixed workloads*
- **Attack coverage**: *accuracy* of an IDS in the presence of *pure malicious workloads*
- **Resistance to evasion techniques**:
  - *overlooked* in comparison to above two, as it was considered to be of limited importance from a practical perspective [17]
  - involves *pure malicious* and *mixed* workloads
- Attack detection and reporting speed: relevant for distributed IDS

Other measurements address performance properties of IDS.

SoTA of the Evaluation of ML/DL-based IDS

Evaluation of an IDS requires:

- a testing environment
- a dataset
- a set of metrics

Evaluation methodologies usually focus on:

- dataset quality
- detection performance metrics
- realistic environment provision

Shortcomings

Most ML/DL-based IDS proposals:

- share the same set of metrics
  - **accuracy** instead of *precision* and *recall*
  - fail to use *MCC* when the dataset is **imbalanced**
- use widespread IDS datasets
  - **KDD99** has been over-used
  - many datasets suffer from **shortcut learning** [22] or labeling errors [23, 24]
- propose comparisons
  - experimental protocols differ, e.g., **tasks are different** (supervised classification vs. anomaly detection)
  - experimental settings differ, e.g., same datasets but **different splits**
  - experiments lack temporal/spatial diversity [18]

Mislabelling in CIC-IDS2017 [23]

- CICFlowMeter issue with misordered packets

Mislabelling in CIC-IDS2017 [23]

- CICFlowMeter issue with misordered packets
  - flow processing happens according to the order of packets in the dataset, not the timestamp

Mislabelling in CIC-IDS2017 [23]

- CICFlowMeter issue with misordered packets
  - flow processing happens according to the order of packets in the dataset, not the timestamp
  - from 0.028 to less than 0.1% frames are misordered resulting in swapped flows

Mislabelling in CIC-IDS2017 [23]

- CICFlowMeter issue with misordered packets
- CIC-IDS2017 contains duplicated packets (up to 13 times)

Mislabelling in CIC-IDS2017 [23]

- CICFlowMeter issue with misordered packets
- CIC-IDS2017 contains duplicated packets (up to 13 times)
  - may be due to port mirroring misconfiguration on the testbed switch

Mislabelling in CIC-IDS2017 [23]

- CICFlowMeter issue with misordered packets
- CIC-IDS2017 contains duplicated packets (up to 13 times)
  - may be due to port mirroring misconfiguration on the testbed switch
  - more than 4.5% of the packets are duplicated per day

Mislabelling in CIC-IDS2017 [23]

- CICFlowMeter issue with misordered packets
- CIC-IDS2017 contains duplicated packets (up to 13 times)
- Further investigation led to the discovery of labeling error

Mislabelling in CIC-IDS2017 [23]

- CICFlowMeter issue with misordered packets
- CIC-IDS2017 contains duplicated packets (up to 13 times)
- Further investigation led to the discovery of labeling error
  - 10s of thousands of port scans were wrongly labeled as benign

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment
  - the specter of lab-only evaluation (pitfall I)
  - traffic generation environment should feature heterogeneous and non-stationary workloads

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment
- Contemporaneity and effectiveness of the attacks

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment
- Contemporaneity and effectiveness of the attacks
  - datasets tend to become rapidly obsolete
  - some attacks are quite ineffective against suitably-configured targets

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment
- Contemporaneity and effectiveness of the attacks
- Representativeness of the normal baselines

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment
- Contemporaneity and effectiveness of the attacks
- Representativeness of the normal baselines
  - normal traffic baseline is crucial
  - problem typically neglected

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment
- Contemporaneity and effectiveness of the attacks
- Representativeness of the normal baselines
- Other concerns

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment
- Contemporaneity and effectiveness of the attacks
- Representativeness of the normal baselines
- Other concerns
  - bugs of the feature extractor leading to incorrect flow records
  - data labeling
  - class imbalance

Datasets: A Nail in the Coffin? [25]

The role of publicly available datasets in advancing NIDS development found to be questionable

- Simplifications of the data collection environment
- Contemporaneity and effectiveness of the attacks
- Representativeness of the normal baselines
- Other concerns (already mentioned earlier!)

The Temptation of Synthetic Legitimate Traffic [26]

Aside from the availability of data due to privacy concerns or neglect

The Temptation of Synthetic Legitimate Traffic [26]

Aside from the availability of data due to privacy concerns or neglect

space the one-size-fits-all dataset does not exist:

The Temptation of Synthetic Legitimate Traffic [26]

Aside from the availability of data due to privacy concerns or neglect

space — the one-size-fits-all dataset does not exist: environments are **specific**

The Temptation of Synthetic Legitimate Traffic [26]

Aside from the availability of data due to privacy concerns or neglect

space
: the one-size-fits-all dataset does not exist: environments are **specific**

time
: the traffic data is assumed to be drawn independently and identically:

The Temptation of Synthetic Legitimate Traffic [26]

Aside from the availability of data due to privacy concerns or neglect

space
: the one-size-fits-all dataset does not exist: environments are **specific**

time
: the traffic data is assumed to be drawn independently and identically: environments are **non-stationary**

The Temptation of Synthetic Legitimate Traffic [26]

Aside from the availability of data due to privacy concerns or neglect

space
: the one-size-fits-all dataset does not exist: environments are **specific**

time
: the traffic data is assumed to be drawn independently and identically: environments are **non-stationary**

Additionally, we shall move away from a reactive stance:

The Temptation of Synthetic Legitimate Traffic [26]

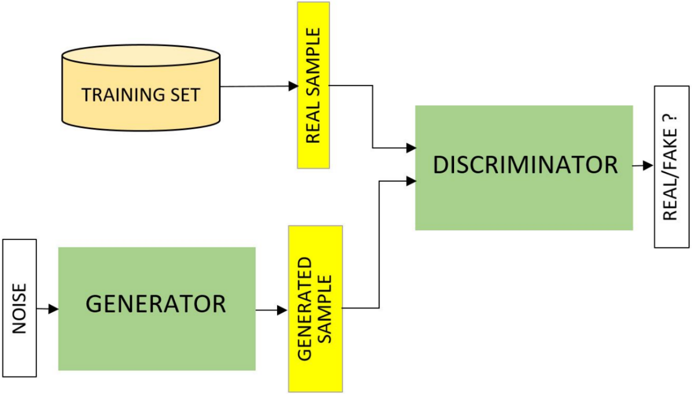
Aside from the availability of data due to privacy concerns or neglect

space
: the one-size-fits-all dataset does not exist: environments are **specific**

time
: the traffic data is assumed to be drawn independently and identically: environments are **non-stationary**

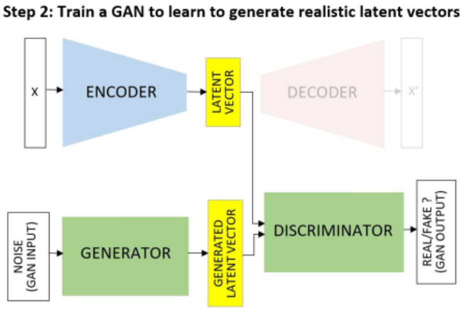
Additionally, we shall move away from a reactive stance: (*new*) attack strategies may be **anticipated**

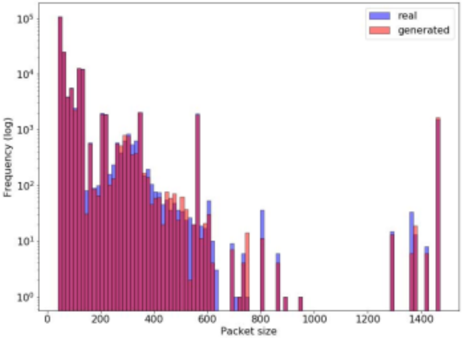
Generative Adversarial Networks (GAN) [28]



GANs are composed of two competing NNs (Figure is courtesy of M.R. Shahid [27])

Learning-based IoT Traffic Generation [27]





Evaluating a Generator [29]

Dataset, although synthetic, still requires a certain level of quality. Since no generally applicable evaluation method was available, we propose our criteria:

- **Realism**: a synthetic sample should be sampled from the same distribution as the real data
- **Diversity**: the distribution of the generated samples should have the same variability as the real data
- **Novelty**: a generated sample should be sufficiently different from the samples of the real distribution
- **Compliance***: generated network traffic must also conform to specifications, standards

Network Traffic Generation Evaluation Framework [29]

	Criterion				Input			Data type	
	Real.	*Div.*	*Nov.*	*Comp.*	*Marg.*	*Cond.*	*Joint*	*Cat.*	*Num.*
JSD	✓	✓			✓			✓	
EMD	✓	✓			✓				✓
CMD	✓					✓		✓	
PCD	✓					✓			✓
Density	✓						✓	✓	✓
Coverage		✓					✓	✓	✓
MD			✓				✓	✓	✓
DKC				✓			✓	✓	✓

- Proposed a BN approach using Hill Climbing with two ways to encode numerical features

Network Traffic Generation Evaluation Framework [29]

	Criterion				**Input**			**Data type**	
	Real.	*Div.*	*Nov.*	*Comp.*	*Marg.*	*Cond.*	*Joint*	*Cat.*	*Num.*
JSD	✓	✓			✓			✓	
EMD	✓	✓			✓				✓
CMD	✓					✓		✓	
PCD	✓					✓			✓
Density	✓						✓	✓	✓
Coverage		✓					✓	✓	✓
MD			✓				✓	✓	✓
DKC				✓			✓	✓	✓

- Proposed a BN approach using Hill Climbing with two ways to encode numerical features
- Compared against GAN-based approaches from the state of the art

Network Traffic Generation Evaluation Framework [29]

	Criterion				Input			Data type	
	Real.	*Div.*	*Nov.*	*Comp.*	*Marg.*	*Cond.*	*Joint*	*Cat.*	*Num.*
JSD	✓	✓			✓			✓	
EMD	✓	✓			✓				✓
CMD	✓					✓		✓	
PCD	✓					✓			✓
Density	✓						✓	✓	✓
Coverage		✓					✓	✓	✓
MD			✓				✓	✓	✓
DKC				✓			✓	✓	✓

- Proposed a BN approach using Hill Climbing with two ways to encode numerical features
- Compared against GAN-based approaches from the state of the art
- Generated data using these approaches for 3 different source datasets

Network Traffic Generation Evaluation Framework [29]

	Criterion				Input			Data type	
	Real.	*Div.*	*Nov.*	*Comp.*	*Marg.*	*Cond.*	*Joint*	*Cat.*	*Num.*
JSD	✓	✓			✓			✓	
EMD	✓	✓			✓				✓
CMD	✓					✓		✓	
PCD	✓					✓			✓
Density	✓						✓	✓	✓
Coverage		✓					✓	✓	✓
MD			✓				✓	✓	✓
DKC				✓			✓	✓	✓

- Proposed a BN approach using Hill Climbing with two ways to encode numerical features
- Compared against GAN-based approaches from the state of the art
- Generated data using these approaches for 3 different source datasets
- Used the framework metrics for to evaluate the generated data

Network Traffic Generation Evaluation Framework [29]

	Criterion				Input			Data type	
	Real.	*Div.*	*Nov.*	*Comp.*	*Marg.*	*Cond.*	*Joint*	*Cat.*	*Num.*
JSD	✓	✓			✓			✓	
EMD	✓	✓			✓				✓
CMD	✓					✓		✓	
PCD	✓					✓			✓
Density	✓						✓	✓	✓
Coverage		✓					✓	✓	✓
MD			✓				✓	✓	✓
DKC				✓			✓	✓	✓

- Proposed a BN approach using Hill Climbing with two ways to encode numerical features
- Compared against GAN-based approaches from the state of the art
- Generated data using these approaches for 3 different source datasets
- Used the framework metrics for to evaluate the generated data
- Used two baselines (source data, data copying approach)

Generation Evaluation Results

	Description	Real data	Naive	BN_{bins}	BN_{GM}	CTGAN	E-WGAN-GP	NetShare
JSD	Realism and Diversity for categorical features (↓)	**0.067**	0.0068	**0.066**	0.070	0.218	0.105	*0.399*
EMD	Realism and Diversity for numerical features (↓)	**0.002**	**0.002**	0.018	0.007	*0.029*	*0.029*	0.003
CMD	Realism of Correlation between categorical features (↓)	**0.037**	0.223	**0.031**	0.040	0.209	0.050	*0.578*
PCD	Realism of Correlation between numerical features (↓)	**0.373**	*1.222*	**0.452**	0.738	0.863	1.219	0.542
Density	Realism of data distribution (↑)	**0.951**	0.355	0.701	**0.855**	0.486	0.702	*0.027*
Coverage	Diversity of data distribution (↑)	**1.000**	0.805	0.792	**0.998**	0.802	0.996	*0.076*
MD	Novelty (=)	**8.692**	7.519	8.312	8.316	7.447	**8.341**	*5.675*
DKC	Compliance (↓)	**0.006**	0.079	0.005	0.005	0.019	**0.004**	*0.129*
Global Rank	Average Ranking (↓)	**1.6**	4.4	3.1	**2.9**	5.1	3.6	*5.8*

- BNs overall better at preserving Realism, Diversity and Compliance

Generation Evaluation Results

	Description	Real data	Naive	BN_{bins}	BN_{GM}	CTGAN	E-WGAN-GP	NetShare
JSD	Realism and Diversity for categorical features (↓)	**0.067**	0.0068	**0.066**	0.070	0.218	0.105	*0.399*
EMD	Realism and Diversity for numerical features (↓)	**0.002**	**0.002**	0.018	0.007	*0.029*	*0.029*	0.003
CMD	Realism of Correlation between categorical features (↓)	**0.037**	0.223	**0.031**	0.040	0.209	0.050	*0.578*
PCD	Realism of Correlation between numerical features (↓)	**0.373**	*1.222*	**0.452**	0.738	0.863	1.219	0.542
Density	Realism of data distribution (↑)	**0.951**	0.355	0.701	**0.855**	0.486	0.702	*0.027*
Coverage	Diversity of data distribution (↑)	**1.000**	0.805	0.792	**0.998**	0.802	0.996	*0.076*
MD	Novelty (=)	**8.692**	7.519	8.312	8.316	7.447	**8.341**	*5.675*
DKC	Compliance (↓)	**0.006**	0.079	0.005	0.005	0.019	**0.004**	*0.129*
Global Rank	Average Ranking (↓)	**1.6**	4.4	3.1	**2.9**	5.1	3.6	*5.8*

- BNs overall better at preserving Realism, Diversity and Compliance
- GANs are less effective in tabular data generation

Generation Evaluation Results

	Description	Real data	Naive	BN_{bins}	BN_{GM}	CTGAN	E-WGAN-GP	NetShare
JSD	Realism and Diversity for categorical features (↓)	**0.067**	0.0068	**0.066**	0.070	0.218	0.105	*0.399*
EMD	Realism and Diversity for numerical features (↓)	**0.002**	**0.002**	0.018	0.007	*0.029*	*0.029*	0.003
CMD	Realism of Correlation between categorical features (↓)	**0.037**	0.223	**0.031**	0.040	0.209	0.050	*0.578*
PCD	Realism of Correlation between numerical features (↓)	**0.373**	*1.222*	**0.452**	0.738	0.863	1.219	0.542
Density	Realism of data distribution (↑)	**0.951**	0.355	0.701	**0.855**	0.486	0.702	*0.027*
Coverage	Diversity of data distribution (↑)	**1.000**	0.805	0.792	**0.998**	0.802	0.996	*0.076*
MD	Novelty (=)	**8.692**	7.519	8.312	8.316	7.447	**8.341**	*5.675*
DKC	Compliance (↓)	**0.006**	0.079	0.005	0.005	0.019	**0.004**	*0.129*
Global Rank	Average Ranking (↓)	**1.6**	4.4	3.1	**2.9**	5.1	3.6	*5.8*

- BNs overall better at preserving Realism, Diversity and Compliance
- GANs are less effective in tabular data generation
- CTGAN particularly prone to *mode invention*

Generation Evaluation Results

	Description	Real data	Naive	BN_{bins}	BN_{GM}	CTGAN	E-WGAN-GP	NetShare
JSD	Realism and Diversity for categorical features (↓)	**0.067**	0.0068	**0.066**	0.070	0.218	0.105	*0.399*
EMD	Realism and Diversity for numerical features (↓)	**0.002**	**0.002**	0.018	0.007	*0.029*	*0.029*	0.003
CMD	Realism of Correlation between categorical features (↓)	**0.037**	0.223	**0.031**	0.040	0.209	0.050	*0.578*
PCD	Realism of Correlation between numerical features (↓)	**0.373**	*1.222*	**0.452**	0.738	0.863	1.219	0.542
Density	Realism of data distribution (↑)	**0.951**	0.355	0.701	**0.855**	0.486	0.702	*0.027*
Coverage	Diversity of data distribution (↑)	**1.000**	0.805	0.792	**0.998**	0.802	0.996	*0.076*
MD	Novelty (=)	**8.692**	7.519	8.312	8.316	7.447	**8.341**	*5.675*
DKC	Compliance (↓)	**0.006**	0.079	0.005	0.005	0.019	**0.004**	*0.129*
Global Rank	Average Ranking (↓)	**1.6**	4.4	3.1	**2.9**	5.1	3.6	*5.8*

- BNs overall better at preserving Realism, Diversity and Compliance
- GANs are less effective in tabular data generation
- CTGAN particularly prone to *mode invention*
- NetShare's invalid data due to failure encoding numerical features correlation

Generation Evaluation Results

	Description	Real data	Naive	BN_{bins}	BN_{GM}	CTGAN	E-WGAN-GP	NetShare
JSD	Realism and Diversity for categorical features (↓)	**0.067**	0.0068	**0.066**	0.070	0.218	0.105	*0.399*
EMD	Realism and Diversity for numerical features (↓)	**0.002**	**0.002**	0.018	0.007	*0.029*	*0.029*	**0.003**
CMD	Realism of Correlation between categorical features (↓)	**0.037**	0.223	**0.031**	**0.040**	0.209	0.050	*0.578*
PCD	Realism of Correlation between numerical features (↓)	**0.373**	*1.222*	**0.452**	0.738	0.863	1.219	**0.542**
Density	Realism of data distribution (↑)	**0.951**	0.355	0.701	**0.855**	0.486	**0.702**	*0.027*
Coverage	Diversity of data distribution (↑)	**1.000**	0.805	0.792	**0.998**	0.802	**0.996**	*0.076*
MD	Novelty (=)	**8.692**	7.519	8.312	**8.316**	7.447	**8.341**	*5.675*
DKC	Compliance (↓)	**0.006**	0.079	**0.005**	**0.005**	0.019	**0.004**	*0.129*
Global Rank	Average Ranking (↓)	**1.6**	4.4	**3.1**	**2.9**	5.1	3.6	*5.8*

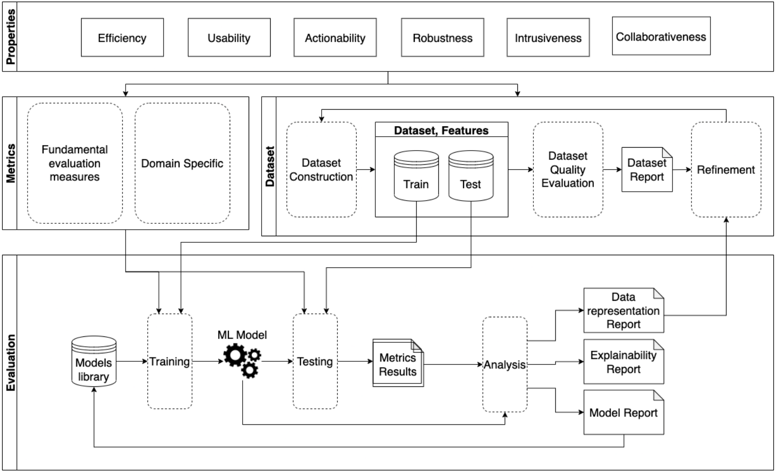
- BNs overall better at preserving Realism, Diversity and Compliance
- GANs are less effective in tabular data generation
- CTGAN particularly prone to *mode invention*
- NetShare's invalid data due to failure encoding numerical features correlation
- BNs more explainable: features' conditional dependency characterizes traffic patterns

Generation Evaluation Results

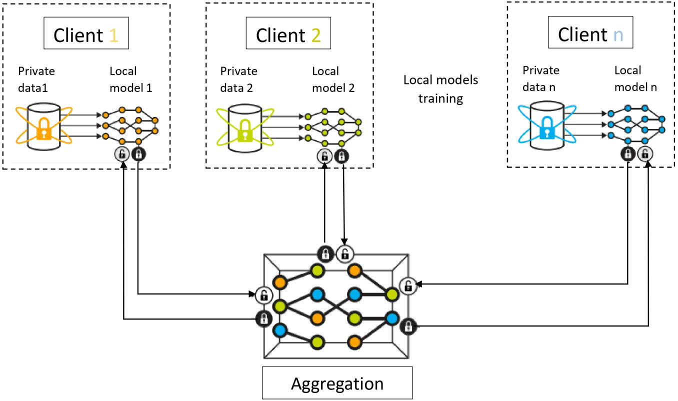
	Description	Real data	Naive	BN_{bins}	BN_{GM}	CTGAN	E-WGAN-GP	NetShare
JSD	Realism and Diversity for categorical features (↓)	**0.067**	0.0068	**0.066**	0.070	0.218	0.105	*0.399*
EMD	Realism and Diversity for numerical features (↓)	**0.002**	**0.002**	0.018	0.007	*0.029*	*0.029*	**0.003**
CMD	Realism of Correlation between categorical features (↓)	**0.037**	0.223	**0.031**	0.040	0.209	0.050	*0.578*
PCD	Realism of Correlation between numerical features (↓)	**0.373**	*1.222*	**0.452**	0.738	0.863	1.219	**0.542**
Density	Realism of data distribution (↑)	**0.951**	0.355	0.701	**0.855**	0.486	0.702	*0.027*
Coverage	Diversity of data distribution (↑)	**1.000**	0.805	0.792	**0.998**	0.802	**0.996**	*0.076*
MD	Novelty (=)	**8.692**	7.519	8.312	8.316	7.447	**8.341**	*5.675*
DKC	Compliance (↓)	**0.006**	0.079	0.005	0.005	0.019	**0.004**	*0.129*
Global Rank	Average Ranking (↓)	**1.6**	4.4	3.1	**2.9**	5.1	3.6	*5.8*

- BNs overall better at preserving Realism, Diversity and Compliance
- GANs are less effective in tabular data generation
- CTGAN particularly prone to *mode invention*
- NetShare's invalid data due to failure encoding numerical features correlation
- BNs more explainable: features' conditional dependency characterizes traffic patterns
- BNs consistently emerge as the most efficient model

Framework for Data-driven NIDS Evaluation [30]



Collaborative Intrusion Detection: Federated Learning



FL offers a way to distribute learning across several clients training local models on private data (Figure is courtesy of S. Chennoufi [31])

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

- Lack of 5G datasets

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

- Lack of 5G datasets
- Datasets are devoid of attack traffic

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

- Lack of 5G datasets
- Datasets are devoid of attack traffic
- Evaluation resorts to using publicly available IDS datasets

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

- Lack of 5G datasets
- Datasets are devoid of attack traffic
- Evaluation resorts to using publicly available IDS datasets
- Evaluation metrics concentrate on accuracy, leaving out FPR

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

- Lack of 5G datasets
- Datasets are devoid of attack traffic
- Evaluation resorts to using publicly available IDS datasets
- Evaluation metrics concentrate on accuracy, leaving out FPR
- Evaluation lacks realism

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

- Lack of 5G datasets
- Datasets are devoid of attack traffic
- Evaluation resorts to using publicly available IDS datasets
- Evaluation metrics concentrate on accuracy, leaving out FPR
- Evaluation lacks realism
  - Validation is done on private test set at the central server

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

- Lack of 5G datasets
- Datasets are devoid of attack traffic
- Evaluation resorts to using publicly available IDS datasets
- Evaluation metrics concentrate on accuracy, leaving out FPR
- Evaluation lacks realism
  - Validation is done on private test set at the central server
  - Training data is randomly distributed

Collaborative Intrusion Detection: Federated Learning

Recent SoK [31] on FL-IDS for 5G demonstrates several evaluation shortcomings

- Lack of 5G datasets
- Datasets are devoid of attack traffic
- Evaluation resorts to using publicly available IDS datasets
- Evaluation metrics concentrate on accuracy, leaving out FPR
- Evaluation lacks realism
  - Validation is done on private test set at the central server
  - Training data is randomly distributed

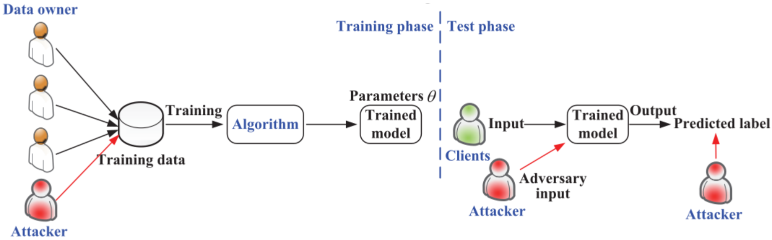
We advocate for more <u>realistic</u> evaluation leveraging **non-IID** data distribution across clients

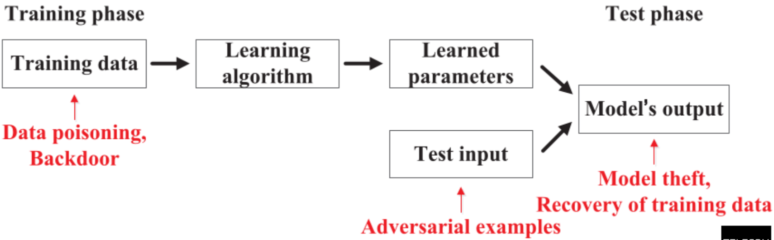
Evaluation of ML-based NIDS: Takeaways

- Lack of a standardized evaluation approach [1]
- Datasets and metrics need to be adapted to the property to assess [30]
- Good quality (legitimate) data is lacking (mostly neglected [25])
- Data, code, hyperparameters are needed to reproduce results [1]
- Baselines are needed to demonstrate the worth of ML/DL [15]
- Comprehensive evaluation is needed in time and space, including unbalanced, non-IID or noisy scenarios

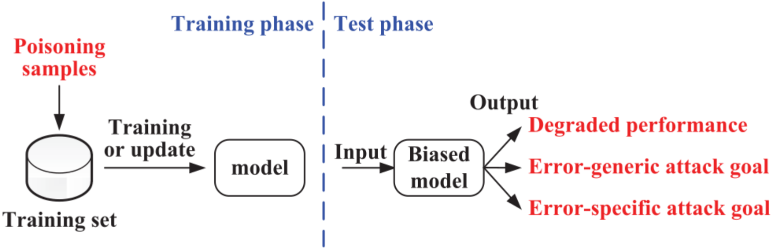
Outline

Threats against ML Systems [32]





Poisoning attacks [32]



Evasion Attacks [33]

Threat model

- Knowledge restriction
  - white box: training dataset and model architecture
  - black box: nothing
- Attack objective
  - untargeted
    - confidence reduction: decrease performance
    - misclassification
  - targeted
    - targeted misclassification: for any input
    - source/target misclassification: for a certain input

Problem formulation

Minimize: $D(x, x + \delta)$ such that:

- $C(x + \delta) = t$ (constraint 1)
- $x + \delta \in [0, 1]^n$ (constraint 2)

Properties of Adversarial Examples [33]

- Perturbation (L_p norms):
- Domain constraints
- Manipulation space:

Properties of Adversarial Examples [33]

- Perturbation (L_p norms): used to compute a minimal pertubation between x and x^{adv}
  - L_0: counts number of modified features
  - L_1: $|x_1 - x_1^{adv}| + \cdots + |x_n - x_n^{adv}|$ (Manhattan distance)
  - L_2: $\sqrt{(x_1 - x_1^{adv})^2 + \cdots + (x_n - x_n^{adv})^2}$ (Euclidean distance)
  - L_∞: $max(|x_1 - x_1^{adv}|, \ldots, |x_n - x_n^{adv}|)$
- Domain constraints
- Manipulation space:

Properties of Adversarial Examples [33]

- Perturbation (L_p norms):
- Domain constraints
  - Syntactic constraints [34]:
    - out-of-range: violations of theoretical bounds (e.g., TTL $>$ 255)
    - binary: violations of the binary nature of a field (e.g., float)
    - multiple category: violation of the one-hot encoding of a field (e.g., both TCP and UDP)
  - Semantic links [35, 36]:
- Manipulation space:

Properties of Adversarial Examples [33]

- Perturbation (L_p norms):
- Domain constraints
  - Syntactic constraints [34]:
  - Semantic links [35, 36]:
    - G0: features related to backward flows (from the server), computed from other features
    - G1: independent features not used to compute other features
    - G2: independent features used to compute other features
    - G3: features dependent on a batch of packets or with underlying physical constraints
- Manipulation space:

Properties of Adversarial Examples [33]

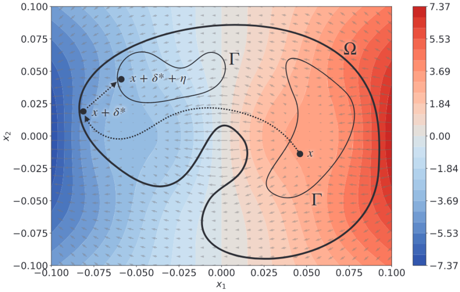
- Perturbation (L_p norms):
- Domain constraints
- Manipulation space:
  - feature-based
  - traffic-based (also known as *problem-based*)

Are Adversarial Examples against NIDS Practical? [34]

Criterion	Value intervals			Non-binary values			Multiple categories		
Dataset	NSL-KDD	UNSW-NB15	CIDDS-01	NSL-KDD	UNSW-NB15	CIDDS-01	NSL-KDD	UNSW-NB15	CIDDS-01
FGSM	100%	100%	100%	100%	100%	100%	100%	100%	100%
BIM	100%	100%	100%	100%	100%	100%	100%	100%	100%
DeepFool	100%	100%	100%	100%	100%	100%	100%	100%	100%
C&WL_2	99.38%	99.55%	99.01%	100%	99.97%	99.92%	0%	0%	0%
C&WL_∞	73.70%	93.15%	98.97%	75.46%	93.38%	99.82%	28.26%	48.83%	0.22%
C&WL_0	70.27%	32.77%	0.43%	58.01%	15.19%	99.74%	0.24%	0.02%	0.48%
JSMA	0.01%	6.52%	0%	31.93%	68.32%	0.67%	31.02%	68.32%	0.67%

Proportion of generated samples violating the practicality criteria.

Feature Space vs. Problem Space [37]



Example of projection of the feature-space attack vector $x + \delta*$ in the *feasible* problem space, resulting in side-effect features η

Towards XAI-driven Adversarial Examples for NIDS [38]

Main objectives

- Problem-space
- Practical
- Decision-driven

Towards XAI-driven Adversarial Examples for NIDS [38]

Main objectives

- Problem-space
- Practical
- Decision-driven

Requirements

- Enumerate valid traffic manipulations and evaluate their impact on feature space
- Design feature selection criteria: *non-functional*, *non-correlated*, manipulated from problem-space, impactful on evasion
- Discover the decision boundary through XAI

XAI-driven Adversarial Perturbation: Method [38]

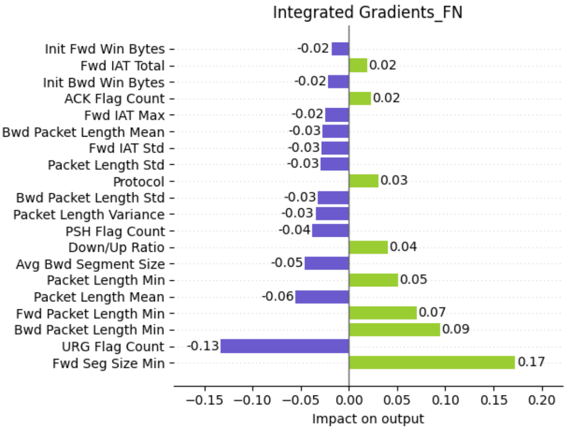
1. Compute *feature importance*, e.g., using Integrated Gradients or SHAP
2. Compute correlation matrix of features
3. Select most important AND less correlated features
4. Plot True Positives and Negatives (e.g., False Negatives) into projected feature space (restricted to the selected features)
5. Evaluate potential feature-space manipulation and choose ones that are possible in problem-space
6. Generate adversarial examples by applying the retained manipulation

XAI-driven Generation: Whitebox Use Case [38]

Step 0: Train a classifier on CIC-IDS2017 dataset and fine-tune it using real attack data generated in a testbed

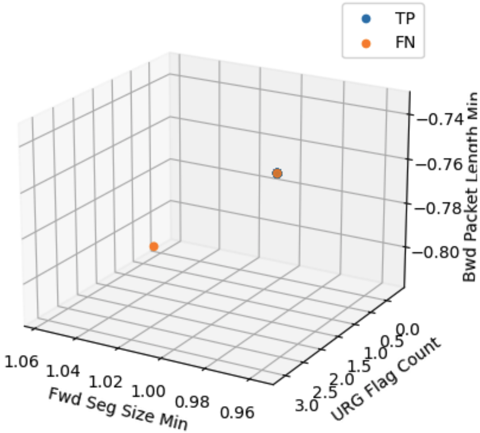
XAI-driven Generation: Whitebox Use Case [38]

Step 1: Compute feature importance using



XAI-driven Generation: Whitebox Use Case [38]

Step 2: Project the TP and FN into the important features space



XAI-driven Generation: Whitebox Use Case [38]

Step 3: Increase `Fwd Seg Size Min` by adding padding to SYN packets

```
tcp_options = [('NOP', None)] * 10
tcp = TCP(sport=random.randint(1, 65535),dport=80,flags='S',seq=100, options=tcp_options)
```

XAI-driven Generation: Whitebox Use Case [38]

Step 4: Manual validation

- generated sample bypasses detection in feature space
- generated sample compromises target in problem space

```
┌──(root㉿kali)-[/home/kali/advex/web_attack/xss]
└─# python http_FwdSegMin.py
Begin emission:
Finished sending 1 packets.
.*
Received 2 packets, got 1 answers, remaining 0 packets
.
Sent 1 packets.
Begin emission:
Finished sending 1 packets.
**.*
Received 4 packets, got 3 answers, remaining 0 packets
b"HTTP/1.1 200 OK\r\nDate: Tue, 02 Jan 2024 15:31:14 GMT\r\nServer: Apache/2.4.6 (CentOS) PHP/5.4.16\r\nX-Powered-By:
 PHP/5.4.16\r\nContent-Length: 143\r\nConnection: close\r\nContent-Type: text/html; charset=UTF-8\r\n\r\nlogin failed
!<br/>select * from users where username='<video controls><source src=validvideo.mp4 type=video/mp4><track default' a
nd password=''"
Begin emission:
Finished sending 1 packets.
.*
Received 2 packets, got 1 answers, remaining 0 packets
.
Sent 1 packets.
```

Adversarial Examples against ML-based NIDS: Takeaway

- DL algorithms are inherently vulnerable to adversarial examples
- Most attack scenarios of the SoTA are unrealistic [39, 1]
- Many approaches from the SoTA are unpractical [34]
- The flow feature extraction function is not invertible in the network traffic domain [37]
- New approaches generating problem-space adversarial examples are emerging but are difficult to evaluate
- Problem-space adversarial examples require exploit-based validation

Outline

Limitations of ML/DL applied to NIDS

- Data labelling approaches towards semi-supervised approaches
- Dataset quality needs to be uniformized
- Evaluation approaches need to be standardized
- Robustness wrt both data dynamics (drifts) and adversarial examples require more practical assessment
- The network flow format has lived: additional indicators are needed to go beyond anomalies
- Need to extract and organize the intrusion knowledge

ML for Cybersecurity: Beyond Threat Detection [1]

- **Alert Management**
  - Alert fusion
  - Alert filtering
  - Alert prioritization
- **Raw-data Analysis**
  - Operational decisions
  - Labelling optimization
- **Risk Exposure Assessment**
  - Penetration testing
  - Compromise indicators
- **Cyber Threat Intelligence**
  - Internal sources
  - External sources

Future works

- NIDS: towards hybrid and knowledge-based model, e.g., provenance graphs, knowledge graphs or GNN-IDS [40]
- evaluation: towards standardized data-driven methodologies
- datasets: towards unified dataset quality metrics, best practices for data generation
- synthetic traffic: towards temporal flow generation
- adversarial examples: towards more realistic attack scenarios, data-driven efficient generation

Thanks for your attention!

https://cloudgravity.github.io

@cloudgravity

gregory.blanc@telecom-sudparis.eu

Advertisement: Postdoc

If you are a Ph.D in Cybersecurity and/or Machine Learning, we have several postdoctoral fellowships (to start ASAP)

- **SuperviZ** Explaining IDS Decisions through Visualisations
- **SuperviZ** Test Data Generation using Traffic Manipulation
- **CKRISP** Human-AI Interaction for Cyberattack Data Generation

Do not hesitate to ask about them or drop me an email!

Advertisement: ARTMAN 2024

2nd Workshop on Recent Advances in Resilient and Trustworthy Machine learning-driveN systems

- co-located with ACSAC 40
- set in Hawaii
- to be held on December 9th
- other important dates:

  Submission Deadline September 1, 2024
  Acceptance Notification October 6, 2024
  Final Manuscript Deadline November 3, 2024



References I

- G. Apruzzese, P. Laskov, E. Montes de Oca, W. Mallouli, L. Brdalo Rapa, A. V. Grammatopoulos, and F. Di Franco, "The role of machine learning in cybersecurity," *Digital Threats: Research and Practice*, vol. 4, no. 1, pp. 1–38, 2023.
- B. Claise, "Cisco Systems NetFlow Services Export Version 9." RFC 3954, Oct. 2004.
- M. Sarhan, S. Layeghy, N. Moustafa, and M. Portmann, "Netflow datasets for machine learning-based network intrusion detection systems," in *Big Data Technologies and Applications: 10th EAI International Conference, BDTA 2020, and 13th EAI International Conference on Wireless Internet, WiCON 2020, Virtual Event, December 11, 2020, Proceedings 10*, pp. 117–135, Springer, 2021.
- M. Sarhan, S. Layeghy, and M. Portmann, "Towards a standard feature set for network intrusion detection system datasets," *Mobile networks and applications*, pp. 1–14, 2022.

References II

M. Sarhan, S. Layeghy, and M. Portmann, “Evaluating standard feature sets towards increased generalisability and explainability of ml-based network intrusion detection,” *Big Data Research*, vol. 30, p. 100359, 2022.

N. Moustafa, J. Hu, and J. Slay, “A holistic review of network anomaly detection systems: A comprehensive survey,” *Journal of Network and Computer Applications*, vol. 128, pp. 33–55, 2019.

M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, “A survey of network-based intrusion detection data sets,” *Computers & security*, vol. 86, pp. 147–167, 2019.

A. L. Buczak and E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” *IEEE Communications surveys & tutorials*, vol. 18, no. 2, pp. 1153–1176, 2015.

References III

- C. F. Pontes, M. M. De Souza, J. J. Gondim, M. Bishop, and M. A. Marotta, “A new method for flow-based network intrusion detection using the inverse potts model,” *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1125–1136, 2021.
- D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, “A survey of deep learning methods for cyber security,” *Information*, vol. 10, no. 4, p. 122, 2019.
- G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, “On the effectiveness of machine and deep learning for cyber security,” in *2018 10th international conference on cyber Conflict (CyCon)*, pp. 371–390, IEEE, 2018.
- M. R. Shahid, G. Blanc, Z. Zhang, and H. Debar, “Anomalous communications detection in iot networks using sparse autoencoders,” in *2019 IEEE 18th international symposium on network computing and applications (NCA)*, pp. 1–5, IEEE, 2019.

References IV

H. Jmila, M. Ibn Khedher, G. Blanc, and M. A. El Yacoubi, “Siamese network based feature learning for improved intrusion detection,” in *Neural Information Processing: 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12–15, 2019, Proceedings, Part I 26*, pp. 377–389, Springer, 2019.

Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, “Kitsune: an ensemble of autoencoders for online network intrusion detection,” *arXiv preprint arXiv:1802.09089*, 2018.

D. Arp, E. Quiring, F. Pendlebury, A. Warnecke, F. Pierazzi, C. Wressnegger, L. Cavallaro, and K. Rieck, “Dos and don’ts of machine learning in computer security,” in *31st USENIX Security Symposium (USENIX Security 22)*, pp. 3971–3988, 2022.

S. Axelsson, “The base-rate fallacy and the difficulty of intrusion detection,” *ACM Transactions on Information and System Security (TISSEC)*, vol. 3, no. 3, pp. 186–205, 2000.

References V

R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *2010 IEEE symposium on security and privacy*, pp. 305–316, IEEE, 2010.

F. Pendlebury, F. Pierazzi, R. Jordaney, J. Kinder, and L. Cavallaro, "{TESSERACT}: Eliminating experimental bias in malware classification across space and time," in *28th USENIX security symposium (USENIX Security 19)*, pp. 729–746, 2019.

B. Biggio and F. Roli, "Wild patterns: Ten years after the rise of adversarial machine learning," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2154–2156, 2018.

P. Mell, V. Hu, R. Lippmann, J. Haines, and M. Zissman, "An overview of issues in testing intrusion detection systems," Tech. Rep. NIST Interagency or Internal Report (IR) 7007, National Institute of Standards and Technology, Gaithersburg, MD, 2003.

References VI

A. Milenkoski, M. Vieira, S. Kounev, A. Avritzer, and B. D. Payne, "Evaluating computer intrusion detection systems: A survey of common practices," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, pp. 1–41, 2015.

L. D'hooge, M. Verkerken, B. Volckaert, T. Wauters, and F. De Turck, "Establishing the contaminating effect of metadata feature inclusion in machine-learned network intrusion detection models," in *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pp. 23–41, Springer, 2022.

M. Lanvin, P.-F. Gimenez, Y. Han, F. Majorczyk, L. Mé, and E. Totel, "Errors in the cicids2017 dataset and the significant differences in detection performances it makes," in *International Conference on Risks and Security of Internet and Systems*, pp. 18–33, Springer, 2022.

References VII

- L. Liu, G. Engelen, T. Lynar, D. Essam, and W. Joosen, "Error prevalence in nids datasets: A case study on cic-ids-2017 and cse-cic-ids-2018," in *2022 IEEE Conference on Communications and Network Security (CNS)*, pp. 254–262, IEEE, 2022.
- M. Catillo, A. Pecchia, and U. Villano, "Machine learning on public intrusion datasets: Academic hype or concrete advances in nids?," in *2023 53rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks-Supplemental Volume (DSN-S)*, pp. 132–136, IEEE, 2023.
- S. Abt and H. Baier, "A plea for utilising synthetic data when performing machine learning based cyber-security experiments," in *Proceedings of the 2014 workshop on artificial intelligent and security workshop*, pp. 37–45, 2014.

References VIII

M. R. Shahid, G. Blanc, H. Jmila, Z. Zhang, and H. Debar, "Generative deep learning for internet of things network traffic generation," in *2020 IEEE 25th Pacific Rim International Symposium on Dependable Computing (PRDC)*, pp. 70–79, IEEE, 2020.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

A. Schoen, G. Blanc, Y. Han, P.-F. Gimenez, F. Majorczyk, and L. Mé, "A tale of two methods: Unveiling the limitations of gan and the rise of bayesian networks for synthetic network traffic generation," in *9th International Workshop on Traffic Measurements for Cybersecurity (WTMC)* (IEEE, ed.), 2024.

References IX

S. Ayoubi, G. Blanc, H. Jmila, T. Silverston, and S. Tixeuil, "Data-driven evaluation of intrusion detectors: a methodological framework," in *International Symposium on Foundations and Practice of Security*, pp. 142–157, Springer, 2022.

S. Chennoufi, G. Blanc, H. Jmila, and C. Kiennert, "(sok) federated learning based network intrusion detection in 5g: Context, state of the art and challenges," in *19th International Conference on Availability, Reliability and Security (ARES)* (ACM, ed.), 2024.

M. Xue, C. Yuan, H. Wu, Y. Zhang, and W. Liu, "Machine learning security: Threats, countermeasures, and evaluations," *IEEE Access*, vol. 8, pp. 74720–74742, 2020.

I. Debicha, B. Cochez, T. Kenaza, T. Debatty, J.-M. Dricot, and W. Mees, "Review on the feasibility of adversarial evasion attacks and defenses for network intrusion detection systems," *arXiv preprint arXiv:2303.07003*, 2023.

References X

- M. A. Merzouk, F. Cuppens, N. Boulahia-Cuppens, and R. Yaich, "Investigating the practicality of adversarial evasion attacks on network intrusion detection," *Annals of Telecommunications*, vol. 77, no. 11, pp. 763–775, 2022.
- M. J. Hashemi, G. Cusack, and E. Keller, "Towards evaluation of nidss in adversarial setting," in *Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks*, pp. 14–21, 2019.
- M. Teuffenbach, E. Piatkowska, and P. Smith, "Subverting network intrusion detection: Crafting adversarial examples accounting for domain-specific constraints," in *Machine Learning and Knowledge Extraction: 4th IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2020, Dublin, Ireland, August 25–28, 2020, Proceedings 4*, pp. 301–320, Springer, 2020.

References XI

- F. Pierazzi, F. Pendlebury, J. Cortellazzi, and L. Cavallaro, "Intriguing properties of adversarial ml attacks in the problem space," in *2020 IEEE symposium on security and privacy (SP)*, pp. 1332–1349, IEEE, 2020.
- S. Okada, H. Jmila, K. Akashi, T. Mitsunaga, Y. Sekiya, H. Takase, G. Blanc, and H. Nakamura, "Xai-driven adversarial attacks on network intrusion detectors," in *European Interdisciplinary Cybersecurity Conference*, pp. 65–73, 2024.
- H. Jmila and M. I. Khedher, "Adversarial machine learning for network intrusion detection: A comparative study," *Computer Networks*, vol. 214, p. 109073, 2022.
- T. Bilot, N. El Madhoun, K. Al Agha, and A. Zouaoui, "Graph neural networks for intrusion detection: A survey," *IEEE Access*, vol. 11, pp. 49114–49139, 2023.