“ Machine Learning Based Cyber Attacks Targeting on Controlled Information: A Survey ”

2022.10.03

Professor: 박종혁

Presented by:

YOTXAY SANGTHONG

Seoul National University of Science and Technology, Seoul, South Korea.

Table Contents

Abstract

- This survey presented the recent advances in this new type of attack and corresponding countermeasures.
- The ML-based stealing attack is reviewed in perspectives of three categories of targeted controlled information.
- Including controlled user activities, controlled ML model-related information, and controlled authentication information.

I. Introduction

- Driven by the needs to protect the enormous value within data and the evolution of the emerging data mining techniques, information leakage becomes a growing concern for governments, organizations and individuals.
- This survey introduced the stealing attack in the cyber security area. The information leakage can be defined as the violation of confidentiality of methods/mechanisms /framework which stores information or has access to information.
- For example, authors [26] extracted user's foreground app running in Android in order to exploit it for the phishing attack, while the user activity information was protected by a nonpublic system level permission.

I. Introduction (Cont.)

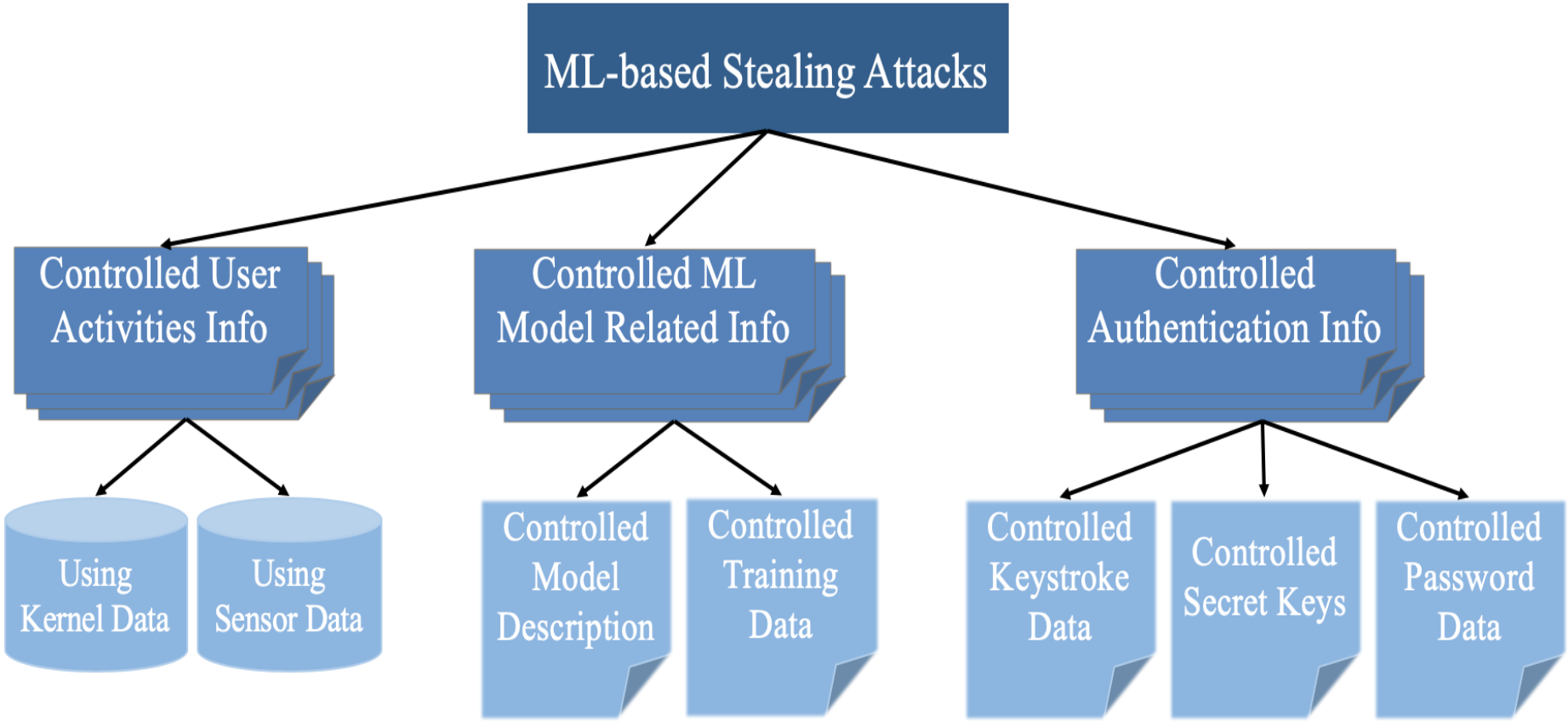


Fig. 1. Introduced Stealing Controlled Information Attack Categories.

I. Introduction (Cont.)

The contributions:

- This paper introduced the ML-based stealing attack, which aims at stealing the controlled/protected information and leads to huge economic loss. ML algorithms are applied in the attack to increase the success rate in various aspects.
- The classification of the ML-based stealing attacks is built based on the targeted controlled information preferentially. Based on this classification, the vulnerabilities in various systems and corresponding attacks are sorted out and revealed.
- The authors surveyed the advances of the ML-based stealing attacks between 2014 and 2019. A methodology applied for the ML-based stealing attack against the controlled information is generalized to five phases reconnaissance, data collection, feature engineering, attacking the objective, and evaluation.
- They discussed the challenges of attacks stealing controlled information and forecast their future directions in terms of how they might affect our digital society.

II. Attack Methodology

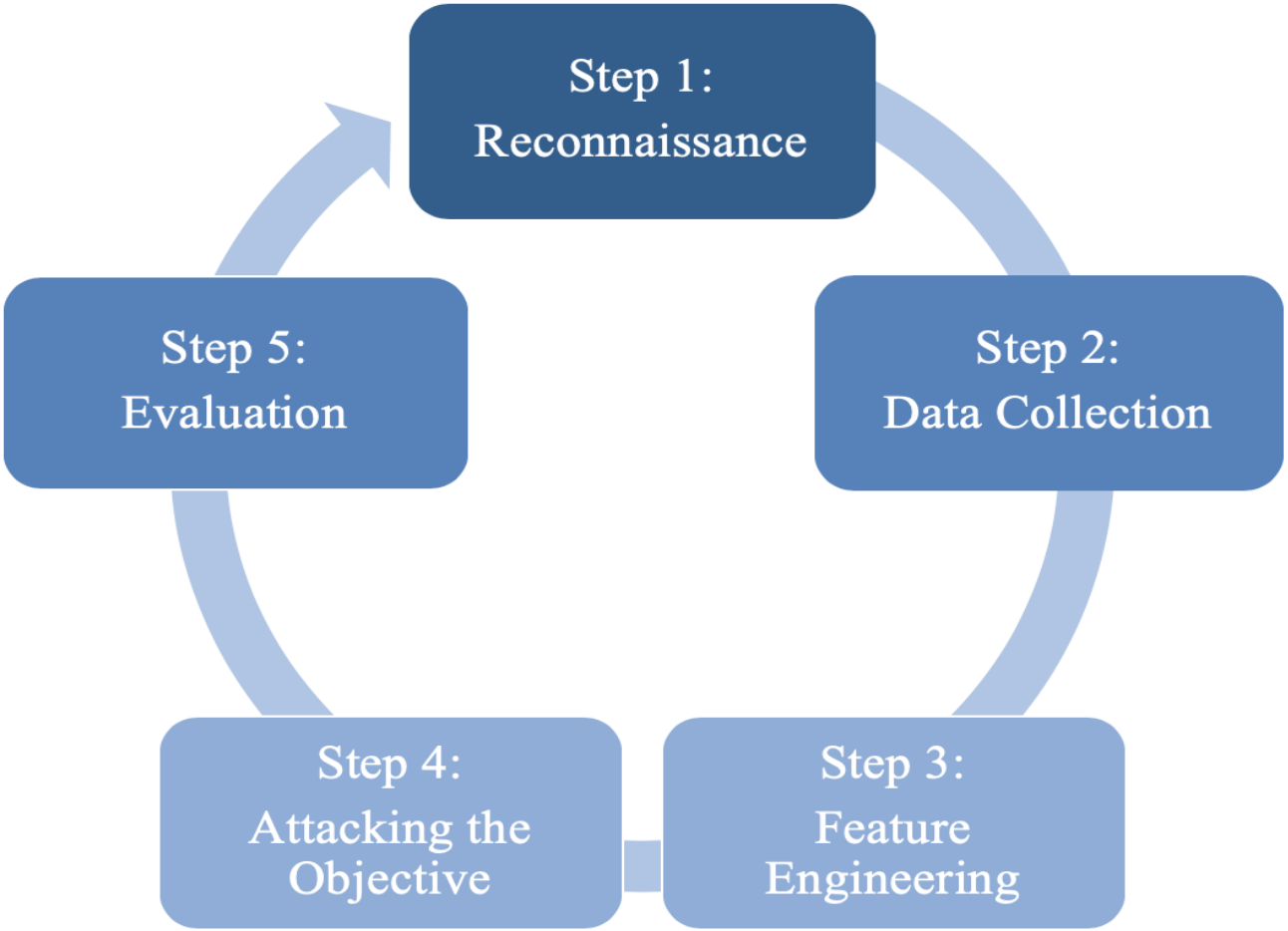


Fig. 2. ML-based stealing attack methodology

II. Attack Methodology (Cont.)

2.1 Reconnaissance

- Reconnaissance refers to a preliminary inspection of the stealing attack. The two aims of this inspection include defining adversaries' targets and analyzing the accessible data in order to facilitate the forthcoming attacks.
- The target of adversaries in the published literature is usually the confidential information controlled by systems and online services.
- The attacker needs to exhaustively search all possible entry points of the targeted system, reachable data paths, and readable data.
- When the attacker aims at user's activities, the triggered hardware devices and their corresponding logged information will be investigated.
- For example, the attacker always searches and explores the readable system files, such as interrupt timing data and network resources.

II. Attack Methodology (Cont.)

2.2 Data Collection

- Active collection refers to the attacker actively interacts with the targeted system for data collection.
- Specifically, an attacker designs some initial queries to interact with the system and subsequently collects the data. The goal of the attacker guides the design of malicious interactions, referring to the analysis results from the reconnaissance phases.
- For example, if an attacker intends to identify which app is launched in a user's mobile, some system files like $proc\ f\ s$ recording app launching activities.

II. Attack Methodology (Cont.)

2.3 Feature Engineering

- After the datasets are prepared, feature engineering is the subsequent essential phase to generate representative vectors of the data to empower the ML model. The two key points in feature engineering for ML-based attacks consist of dataset cleaning and extracting features.

- An obstacle of feature engineering is cleaning the noises and irrelevant information in the raw data. In general, deduplication and interpolation can be used to reduce the noise from accessible resource.

- To reduce the noise, a Fast Fourier Transform (FFT) filter and an Inverse FFT (IFFT) filter are applied.

II. Attack Methodology (Cont.)

2.4 Attacking the Objective

- The ML-based stealing attack into two attack modes as illustrated in Fig. 3. The five actions correspond to the first three phases within the MLBSA methodology.

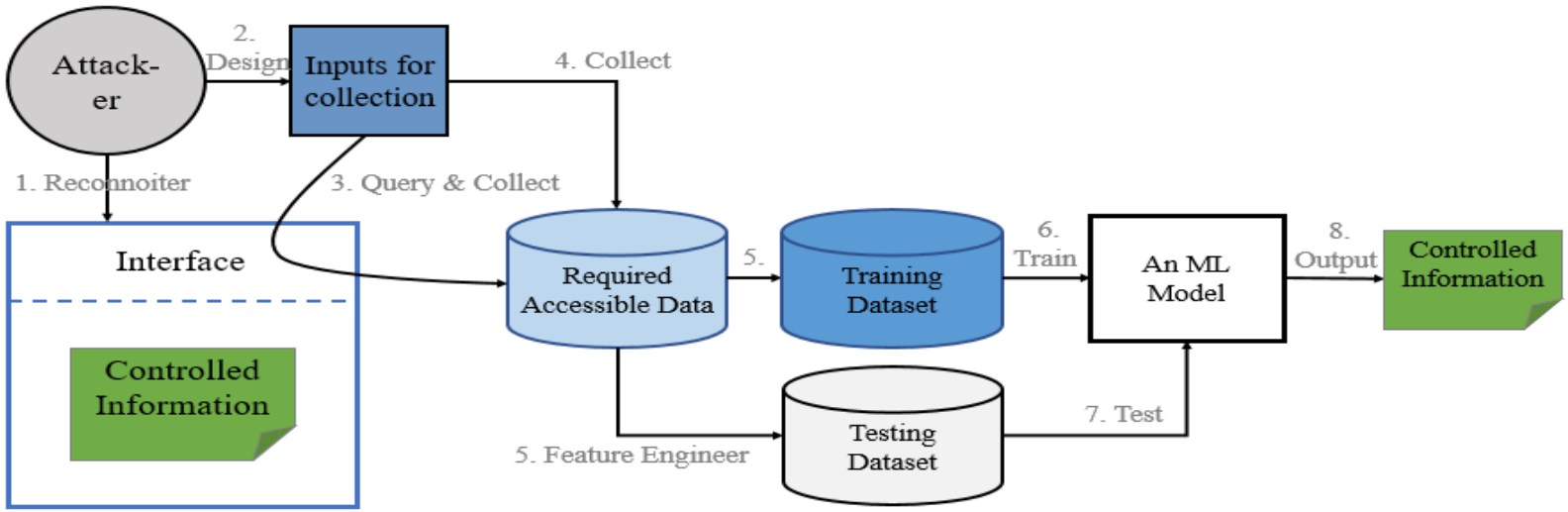
- As stated in the data collection phase, the inputs and their query results are collected as the required accessible dataset, which reveals the target information.

- Based on the target information, the ground truth of the dataset is set up in this phase. With proper feature engineering methods, the training dataset is prepared to attack the objective.

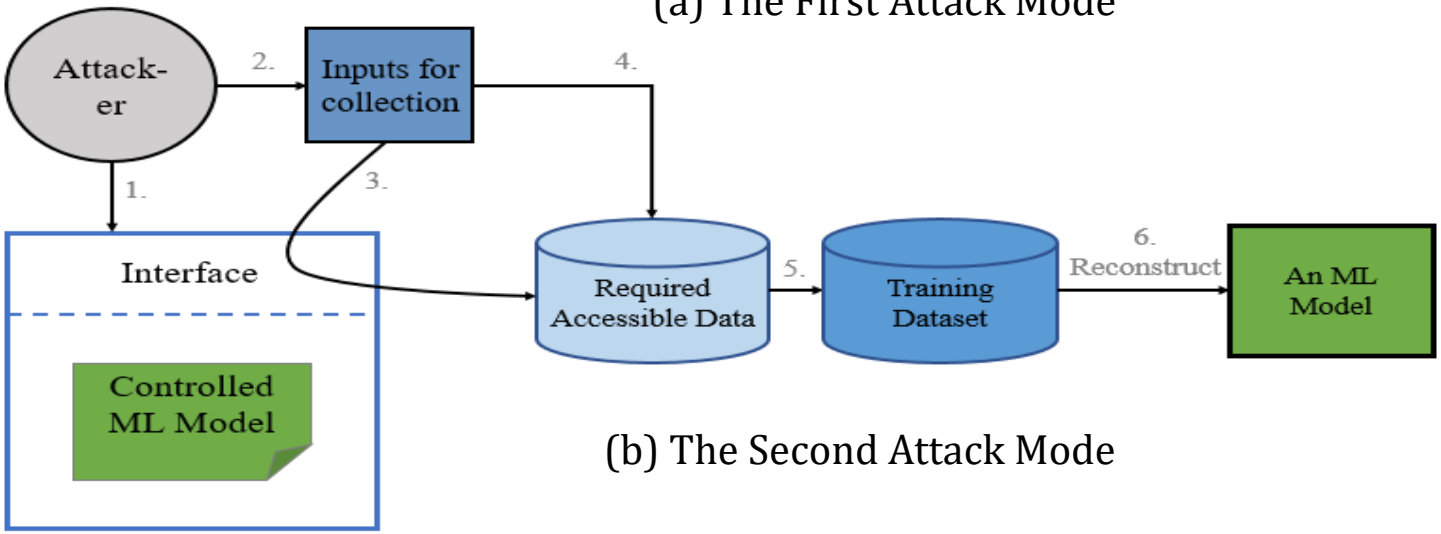
- But the subsequent actions to steal the controlled information using machine learning differ between two attack modes.

II. Attack Methodology (Cont.)

2.4 Attacking the Objective (Cont.)



(a) The First Attack Mode



(b) The Second Attack Mode

Fig. 3. The ML-based stealing attack into two attack modes

II. Attack Methodology (Cont.)

2.4 Attacking the Objective (Cont.)

- For the first attack mode as shown in Fig. 3a, this attack mode is applied in the ML-based stealing attack against the user activity information, the authentication information, and training set information.

- Regarding the testing dataset is collected from a victim's system/service, the testing samples are not labeled while querying the attack model.

- Since the attack model is built to infer the controlled information from these accessible data, the output of the model is the targeted controlled information.

- The second attack mode illustrated in Fig. 3b, is mostly applied against the ML model-related information. In a black-box setup, stealing the ML model attack aims at calculating the detailed expression of the model's objective function.

- Reconstructing the original model is essentially a reconstruction attack. Using the equation-solving and path-finding methods, the inputs and their query outputs for solving the specific objective function expression is interpreted as the training set.

- Therefore, this attack can be simplify regarded as an ML-based attack. Additionally, based on the attackers' inputs and the query outputs, the training set is synthesized and used to build a substitute model for reconstruction.

II. Attack Methodology (Cont.)

2.5 Evaluation

- Evaluation metrics differ between two attack modes.
- For the first attack mode, the attack model is the attacker's weapon.
- Most metrics commonly used to measure the effectiveness include accuracy, precision, recall, FPR, FNR, and F-measure.

- **Accuracy:** It is also known as success rate and inference accuracy. Accuracy means the number of correctly inferred samples to the total number of predicted samples. Accuracy is a generic metric evaluating the attack model's effectiveness.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

- **Precision:** It is regarded as one of the standard metrics for attack accuracy. Precision illustrates the percentage of samples correctly predicted as controlled class *A* among all samples classified as *A*.

$$Precision = \frac{TP}{TP+FP}$$

- **Recall:** It is regarded as another standard metric for attack accuracy. Recall is also called sensitivity or True Positive Rate (TPR). It is the probability of the amount of class A correctly predicted as class *A*. Similar to precision, recall also reveals the model's correctness on a specific class. These two metrics are almost always applied together.

$$Recall = \frac{TP}{TP+FN}$$

II. Attack Methodology (Cont.)

2.5 Evaluation (Cont.)

- **F-measure:** This metric or **F1-score** is the harmonic mean of recall and precision. F-measure provides a comprehensive analysis of precision and recall .

$$F - measure = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

- **False positive rate (FPR):** This metric denotes the proportion of class *B* samples mistakenly categorized as class *A* sampled.

FPR assesses the model's misclassified samples. $FPR = \frac{FP}{TN+FP}$

- **False negative rate (FNR):** This metric stands for the ratio between class *A* samples mistakenly categorized as class *B* samples. Similar to FPR, FNR assesses the model's misclassified samples from another aspect. FPR and FNR are almost always applied together to measure the model's error rate.

$$FNR = \frac{FN}{TP+FN}$$

- **Execution time:** The execution time is used in training the model which indicates the efficiency of the attack model.

- **Battery consumption:** It is also known as **power consumption**. Battery consumption refers to the target mobile's battery while the target system is a mobile system, which indicates the efficiency of the attack model. For the second attack.

II. Attack Methodology (Cont.)

2.5 Evaluation (Cont.)

- For the second attack mode, ML-based attacks of stealing the ML model are assessed with other metrics.

The applied evaluation metrics are defined and listed below:

- **Test error** is the average error based the same test set (D) testing at learned model and targeted model [125]. A low test error means $\hat{f}$ matches f well.

$$Error_{test}(f, \hat{f}) = \frac{\sum_{x \in D} diff(f(x), \hat{f}(x))}{|D|}$$

- **Uniform error** is an estimation of the portion of full feature space that the learned model is different from the targeted one, when the testing set (U) are selected uniformly [125].

$$Error_{uniform}(f, \hat{f}) = \frac{\sum_{x \in U} diff(f(x), \hat{f}(x))}{|U|}$$

- **Extraction accuracy** indicates the performance of model extraction attack based on the test error and the uniform error [125].

$$Accuracy_{extraction} = 1 - Error_{test}(f, \hat{f}) = 1 - Error_{uniform}(f, \hat{f})$$

II. Attack Methodology (Cont.)

2.5 Evaluation (Cont.)

- **Relative estimation error (EE)** measures the effectiveness of model extraction attack using its learned hyperparameters (ˆ λ) contrasting to the original hyperparameters (λ) [131].

$$Error_{EE} = \frac{|\hat{\lambda} - \lambda|}{\lambda}$$

- **Relative mean square error (MSE)** measures how well the model extraction attack reconstructs the regression models via comparing the mean square error after learning hyperparameters using cross-validation techniques [131].

$$Error_{MSE} = \frac{|MSE_{\hat{\lambda}} - MSE_{\lambda}|}{MSE_{\lambda}}$$

- **Relative accuracy error (AccE)** measures how well the model extraction attack reconstructs the classification models via comparing accuracy error after learning hyperparameters using cross-validation techniques [131].

$$Error_{AccE} = \frac{|AccE_{\hat{\lambda}} - AccE_{\lambda}|}{AccE_{\lambda}}$$

III. ML-based Stealing Attacks and Protections

2.5 Evaluation (Cont.)

Table 2. Outline of Reviewed Papers (info: information)

Reference	Year	Targeted Info	Accessible Data	Goals
[26]	2016	Unlock pattern; Foreground app	Hardware interrupt data	Unlock pattern & foreground app inference attacks via analyzing interrupt time collected from interrupt log file.
[119]	2018	Visited websites; Foreground app	Interrupt data; Network &Memory process record	Search and attack the kernel records leaking user's specific events (i.e. app starts, website launch, keyboard gesture).
[151]	2018	Visited websites; Foreground app; Map	Memory data; Network source; File system data	Several side-channel inference attack on iOS mobile device.
[136]	2015	Visited websites; Input keystrokes	Kernel data-structure fields	Protect by injecting noise into the value of kernel data structure values to secure *procfs*.
[50]	2016	Manufacturing activities	Acoustic sensor data; Magnetic sensor data	An attack capture acoustic & magnetic sensor data to steal a manufacturing process specification or a design.
[117]	2017	User activities info	Sensor data	Contextual model detect malicious behavior of sensors like leaking.
[125]	2016	Parameters of an ML model	Input features & Query outputs	Model extraction attacks leverage confidence info with predictions against MLaaS APIs in black-box setting.
[101]	2017	Internal info of an ML model	Input features & Query outputs	Build a local model to substitute the target model and use it craft adversarial examples in black-box setting.
[131]	2018	Hyperparameters of an ML model	Input features & Query outputs	Hyperparameters stealing attack via observing minima objective function against MLaaS in black-box setting.
[98]	2018	Hyperparameters of an ML model	Input features & Query outputs	Build a metamodel to predict hyperparameters with a given classifier in black-box setting to generate adversarial examples.
[34]	2015	Training data for an ML model	Input features & Query outputs & model structure	Model inversion attacks used confidence info leaking training samples with predictions against MLaaS in two settings.
[49]	2017	Training data for an ML model	Input features & Query outputs & model structure	Online Attack using GAN against collaborative deep learning model leaking user's training sample.
[116]	2017	Training data for an ML model	Input features & Query outputs	Membership inference attacks use shadow training technique to leak the specific record's membership of original training set.
[110]	2019	Training data for an ML model	Input features & Query outputs	Enlarge the scope of membership inference attacks by releasing some key assumptions.
[38]	2018	The property of training set	Input features & Query outputs & model structure	Infer global properties of the training data unintended to be shared in white-box setting.

III. ML-based Stealing Attacks and Protections (Cont.)

2.5 Evaluation (Cont.)

Table 2. Outline of Reviewed Papers

[91]	2019	The property of training set	Input features & Query outputs & model structure	Membership inference attacks against collaborative deep learning model leaking others' unintended feature.
[95]	2018	Training data for an ML model	Input features & Query outputs	Protect against black-box membership inference attack using an adversarial training algorithm.
[100]	2017	Training data for an ML model	Input features & Query outputs	Protect training set of model from leakage with teacher and student models using PATE.
[70]	2017	Training data for an ML model	N/A	Protect training dataset in stored from leakage before training.
[79]	2015	Input PINs; User input texts	Acoustic sensor data; Accelerometer data	Attack infers users' inputs on keyboards via accelerometer data within user's smartwatch.
[122]	2016	Input PINs; User input texts	Audio sensor data	Attack infers a user's typed inputs from surreptitious video recordings of a tablet's backside motion.
[42]	2018	Cryptographic keys	TLB Cache data	TLBleed attack TLBs to leak secret keys about victim's memory activities via reversing engineer and ML strategies.
[152]	2016	Secret keys	CPU Cache data	Mitigate access-driven side-channel attacks with CacheBar managing memory pages cacheability.
[132]	2016	Password info	PII & leaked password & site info	Attack with seven mathematical guessing models for seven password guessing scenario using different personal info.
[128]	2014	Password info	Corpus & Site leaked list	Password guessing attack by analyzing its semantic patterns.
[90]	2016	Password info	Corpus library	Mitigate against password guessing attack by modeling password guessability in password creation stage.

III. ML-based Stealing Attacks and Protections (Cont.)

3.1 Controlled User Activities Information

- It is essential for security specialists to protect user activities information. Not only because the private activities are valuable to adversaries.

- Also the adversary can exploit some specific activities to perform malicious attacks such as the phishing attack.

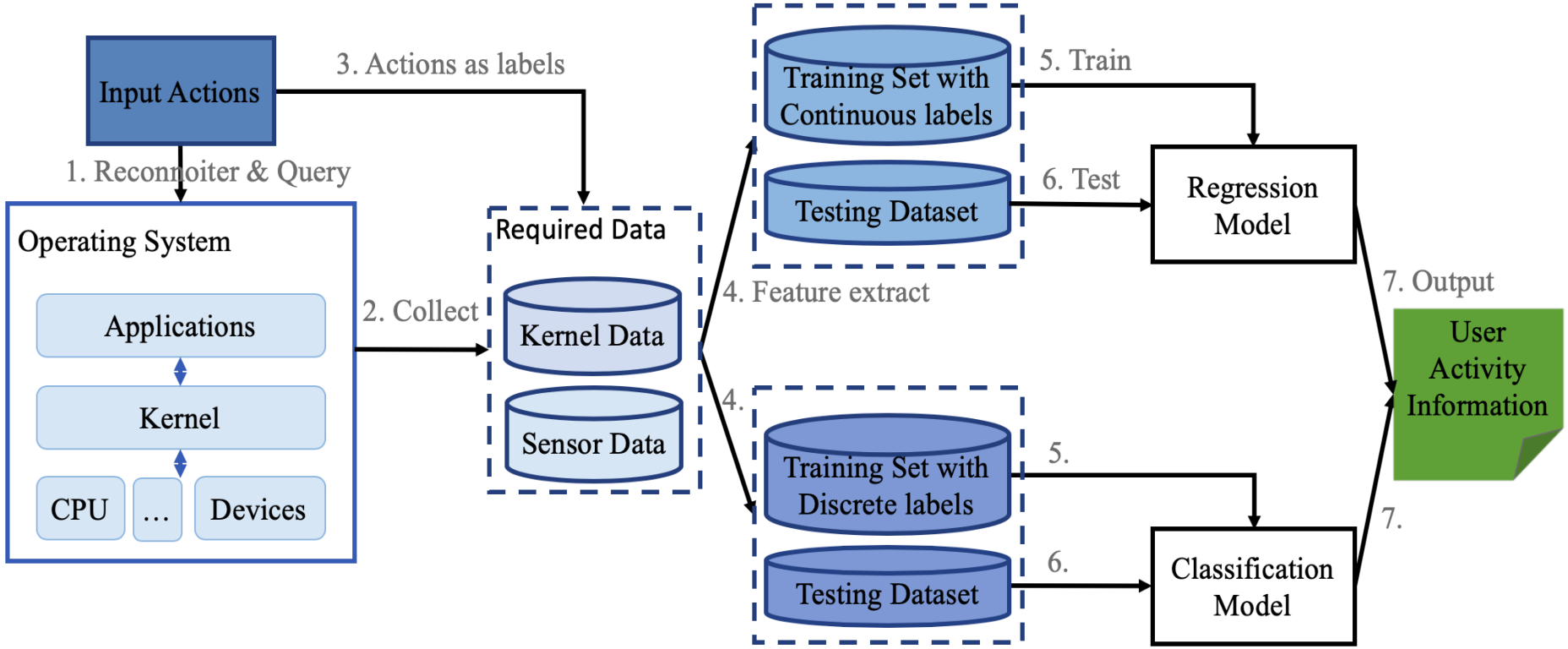


Fig. 4. The ML-based stealing attack against user activities information.

III. ML-based Stealing Attacks and Protections (Cont.)

3.1.1 Stealing controlled user activities from kernel data.

- The dataset collected from the kernel about system process information is too noisy and coarse-grained to disclose any intelligible and valuable information.
- However, through analyzing plenty of such data, the adversary could deduce some confidential information about the victim's activities with the help of ML algorithms.
- ***Stealing User Activities with Timing Analysis:*** The security implications of the kernel information through integrating some specific hardware components into Android smartphones.
- The targeted user activities were unlock patterns and foreground apps. Moreover, users' browsing behavior was targeted by the attacker.

Table 3. Stealing Controlled User Activities using Kernel Data

Reference	Dataset for Experiment	Description	Feature Engineering	ML-based Attack Method
[26]	Interrupt data for unlock pattern and for apps	Collect from *procfs*	Deduplication; Interpolation; Interrupt Increment Computation; Gram Segmentation; DTW	HMM with Viterbi algorithm; k-NN classifier with DTW
[119]	Time series for apps, website,keyboard guests	Collect from *procfs*	Automatically extract with *tsfresh*; DTW	Viterbi algorithm with DTW; SVM classifier with DTW
[151]	1200 x 6 time series of data about app; 1000 website traces	120 apps(App Store+iOS) +10 trace x 6 time series; 10 traces for each website	Manually defined; SAX, BoP representation	SVM classifier; k-NN classifier with DTW
[136]	Consecutively reading data; Resident size field data	Collect from *procfs*	N/A; Construct a histogram binned into seven equal-weight bins	SVM classifiers

III. ML-based Stealing Attacks and Protections (Cont.)

3.1.1 Stealing controlled user activities from kernel data (Cont.).

- ***Stealing User Activities with iOS Side-channel Attack:*** In iOS systems, one popular side-channel attack vector of Linux system about the process information is inaccessible.
- ***Protection using Privacy Mechanism:*** An attack exploiting the kernel process information via decreasing the data's resolution was defended. A differential privacy (DP) mechanism was utilized to prevent the attackers from gaining any useful storage information.

3.1.2 Stealing controlled user activities using sensor data.

- The stealing attack using sensor data should be studied seriously by the defenders, not only from the application of effective ML mechanisms, but also from the popularity of sensing enabled applications.
- The sensor information can reveal the controlled information indirectly as demonstrated in this stealing attack, such as acoustic and magnetic data.

III. ML-based Stealing Attacks and Protections (Cont.)

3.1.2 Stealing controlled user activities using sensor data (Cont.)

Table 4. Stealing Controlled User Activities using Sensor Data

Reference	Dataset for Experiment	Description	Feature Engineering	ML-based Attack Method
[50]	Audio signature dataset	Recorded with a phone put within 4 inches of the printer	STFT, noise normalization	A regression model
[117]	Sensor dataset	Sensor data collected benign and malicious activities	N/A	Markov Chain, NB, LMT, (alternative algorithms e.g. PART)

- ***Stealing Machine's Activities with Sensor-based Attack:*** A side-channel attack was proposed manufacturing equipment exploiting sensor data collected by mobile phones, which revealed its design and the manufacturing process.

- The attacker managed to reconstruct the targeted equipment. As a result of reconnaissance, the security threat of the manufacturing sector was indicated.

- After the dataset was gathered, the ML-based attack was completed by feature engineering, attacking with model training, and evaluation.

III. ML-based Stealing Attacks and Protections (Cont.)

3.1.3 Summary.

- ML-based attacks steal user activities information from operating systems. According to the data sources, there are two kinds of attacks using kernel data and using sensor data.
- Kernel data reveals some system-level behaviors of the target system, while sensor data reflects the system's reactions on its specific functionality used by users [26]. The kernel data is analyzed by the adversary from a time dimension.
- ***Countermeasures:*** Regarding the protection mechanism, differential privacy is an important method for the attacks stealing user activities information.
- The in-depth research in protecting against user activities information can explore the differential privacy appliance or a management system design for kernel files and sensor data.
- Noise injection and access restriction are two effective protections, and the detection can alert the stealing attack.

III. ML-based Stealing Attacks and Protections (Cont.)

3.2 Controlled ML Model Related Information

- ML model related information consists of the model description, training data information, testing data information, and testing results.
- The model description and training data information are controlled, otherwise, it is easy for an attacker to interpret the victim's query result.
- The generalized attack in this category is illustrated in Figure 5. In this category, ML-based attacks aim at stealing the training samples or the ML model.
- Stealing the controlled training sample attacks use an ML model to determine whether the input sample is contained in the target training set.

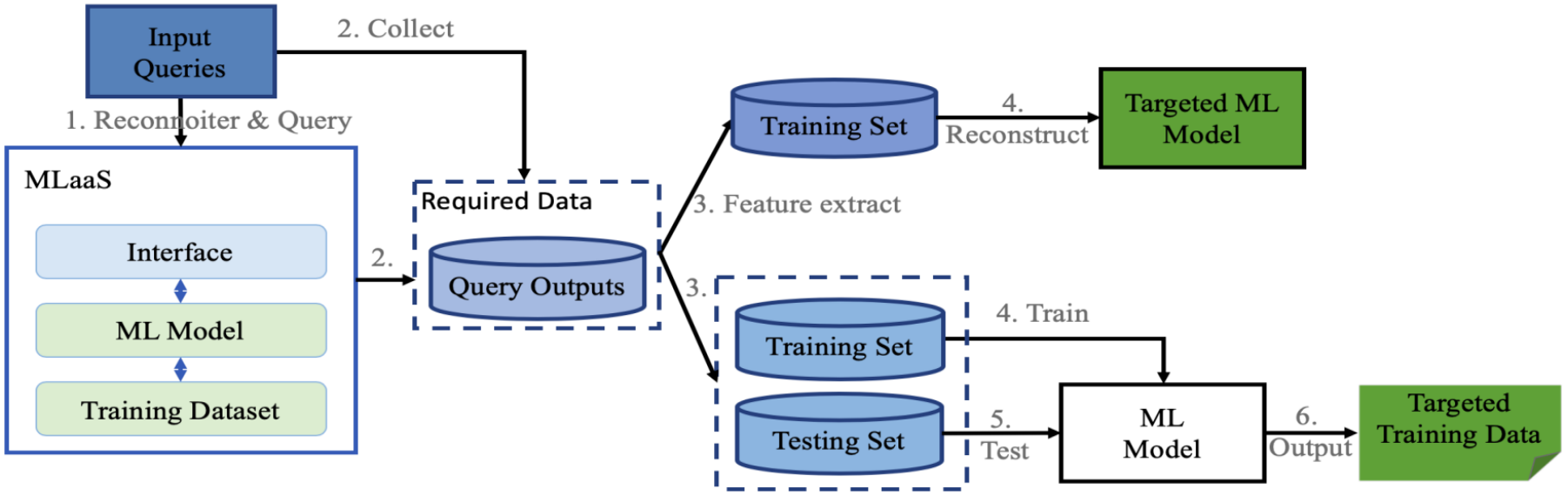


Fig. 5. The ML-based stealing attack against ML model related information.

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.1 Stealing controlled ML model description.

- It is important to protect the confidentiality of ML models online. If fraud detection are based on ML models then understanding the model means that adversaries can evade detection.

- A specific ML model is defined by two important elements including ML algorithm's parameters and hyperparameters. Since the model is controlled, its parameters and hyperparameters should be deemed confidential by nature.

Table 5. Stealing Controlled ML Model Description

Reference	Dataset for Evaluation	Description	Targeted ML Model	Attack Methods
[125]	Circles, Moons, Blobs, 5-Class [125]; Steak Survey [126], GSS Survey [118], Adult (Income/race) [126], Iris [126], Digits [107], Breast Cancer [126], Mushrooms [126], Diabetes [126]	Synthetic, 5,000 with 2 features, Synthetic,1000 with 20 features, 331 records with 40 features, 16,127 records with 101 features, 48,842 records with 108/105 features, 150 records with 4 features, 1,797 records with 64 features, 683 records with 10 features, 8,124 records with 112 features, 768 records with 8 features	Logistic Regression; Decision Tree; SVM; Three-layer NN	Equation-solving attack; Path-finding attack
[101]	MNIST [69], GTSRB [121]	70,000 handwritten digit images, 49,000 traffic signs images	DNN; SVM; k-NN; Decision Tree; Logistic Regression	Jacobian-based Dataset Augmentation
[131]	Diabetes [126], GeoOrig [126], UJIIndoor [126]; Iris [126], Madelon [126], Bank [126]	442 records with 10 features, 1,059 records with 68 features, 19,937 records with 529 features; 100 records with 4 features; 4,400 records with 500 features; 45,210 records with 16 features	Regression algorithms; Logistic regression algorithms; SVM; NN	Equation solving
[98]	MNIST [69]	70,000 handwritten digit images	NNs	Metamodel methods

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.2 Stealing controlled ML model's training data.

- Another type of controlled information about MLaaS product is the training data. Training data is not only useful to construct the model using ML algorithms provided by an MLaaS platform,

- Also sensitive as the records can be private information [34, 35]. Hence, the confidentiality of the model's training data should be protected.

- ***Model Inversion Attack & Defense:*** The model inversion attack was developed via conducting the commercial MLaaS APIs and leveraging confidence information with predictions.

- However, the attack aimed to be applicable across both white-box setting and black-box setting. For the white-box setting, an adversarial client had a prior knowledge about the description of the model as the APIs allowed.

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.2 Stealing controlled ML model's training data (Cont.)

Table 6. Stealing Controlled ML Model's Training Data.

Reference	Dataset for Experiment	Description	Feature Engineering	ML-based Attack Method
[34]	FiveThirtyEight survey, GSS marital happiness survey	553 records with 332 features, 16,127 records with 101 features	N/A	Decision Tree, Regression model
[49]	MNIST [69], AT&T [111]	70,000 handwritten digit images, 400 personal face images	Features learned with DNN	Convolutional Neural Network (CNN) with GAN
[116]	CIFAR10 [65], CIFAR100 [65], Purchases [52], Foursquare [140], Texas hospital stays [47], MNIST [25], Adult (income) [126]	6,000 images in 10 classes, 60,000 images in 100 classes, 10,000 records with 600 features, 1,600 records with 446 features, 10,000 records with 6170 features, 10,000 handwritten digit images, 10,000 records with 14 attribute	Regarded shadow model resulted as features and label records as in/out	NN
[110]	Include 6 sets in [116], News [53], Face [68]	Same as above cell, 20,000 newsgroup documents in 20 classes, 13,000 faces from 1,680 individuals	Regarded shadow model resulted as features and label records as in/out	Random Forest, Logistic Regression, Multilayer perceptron
[38]	Adult (income) [126], MNIST [69], CelebFaces Attributes [80], Hardware Performance Counters	299,285 records with 41 features, 70,000 handwritten digit images, more than 200K celebrity images, 36,000 records with 22 features	Neuron sorting , Set-based representation	NN
[91]	Face [68], FaceScrub [96], PIPA [149], Yelp-health, Yelp-author [141], FourSquare [140], CSI corpus [129]	13,233 faces from 5,749 individuals, 76,541 faces from 530 individuals, 60,000 photos of 2,000 individuals, 17,938 reviews, 16,207 reviews, 15,548 users in 10 locations, 1,412 reviews	N/A	Logistic regression, gradient boosting, Random Forests
[95]	CIFAR100 [65], Purchase100 [52], Texas100 [47]	60,000 images in 100 classes, 197,324 records with 600 features, 67,330 records with 6,170 features	Regarded shadow model resulted as features and label records as in/out	NN

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.2 Stealing controlled ML model's training data (Cont.)

- ***Stealing the Training Data of Deep Model with GAN:*** An attack against the privacy-preserving collaborative deep learning was designed to leak the participants' training data which might be confidential.
- A distributed, federated, or decentralized deep learning algorithm can process each users' training set by sharing the subset of parameters obfuscated with differential privacy.
- However, the training dataset leakage problem had not been solved by using the collaborative deep learning model.
- ***Membership Inference Attack:*** Learning a specific data record which was the membership of the training set of the targeted MLaaS model.
- Since the commercial ML model only allowed black-box access provided by Google and Amazon, not only the training data but also the training data's underlying distribution were controlled.
- Therefore, an attack model was trained which could recognize such differences and determine whether the input data was the member of targeted training set or not. The attack is intended to recognize the model's behavior testing with target training sample.

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.2 Stealing controlled ML model's training data (Cont.)

- ***Property Inference Attack:*** Different from learning a specific training record, the property inference attack targets at the properties of training data that the model producer unintended to share.
- The target model was defined as a white-box Fully Connected Neural Networks (FCNNs) with the aim to infer some global properties such as a higher proportion.
- During the feature engineering phase, the meta-training set by applying set-based representation instead of using a flattened vector of all parameters [5].

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.2 Stealing controlled ML model's training data (Cont.)

Table 7. Categories of Stealing ML related information attacks from three perspectives.

Attack Type	Attack Targets		Attack Surfaces		Attacker's Capabilities	
	Model Info	Training Set Info	Training Phase	Inference Phase	Black-box Access	White-box Access
Model extraction attack [125]	YES	no	no	YES	YES	no
Model extraction attack [101]	YES	no	no	YES	YES	no
Hyperparameter stealing attack [131]	YES	no	no	YES	YES	no
Hyperparameter stealing attack [98]	YES	no	no	YES	YES	no
Black-box inversion attack [34]	no	YES	no	YES	YES	no
White-box inversion attack [34]	no	YES	no	YES	no	YES
GAN attack [49]	no	YES	YES	no	no	YES
Membership inference attack [116]	no	YES	no	YES	YES	no
Membership inference attack [110]	no	YES	no	YES	YES	no
Property inference attack [38]	no	YES	no	YES	no	YES
Property inference attack [91]	no	YES	YES	no	no	YES

- As for attack targets, two types of information may be stolen model internal information and training set information. From attack surfaces, attacks may occur during either model's training phase or inference phase.

- Considering the attacker's capability, the ML model usually allows either the black-box access or the white-box access. The first category is used for this subsection's organization.

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.2 Stealing controlled ML model's training data (Cont.)

Table 8. Attack's prior knowledge under black-box access and white-box access.

Model's Information	Black-box Access	White-box Access
Predicted Label	YES	YES
Predicted Confidence	YES	YES
Parameters	NO	YES
Hyperparameters	NO	YES

- The black-box access allows the users to query the model and obtain prediction outputs which include the predicted label and confidence value.
- The white-box access allows the users to access any information of its model which includes predicted label, predicted confidence, parameters, and hyperparameters.

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.3 Summary.

- ML-based stealing attacks against model related information target at either model descriptions or model's training data.
- Attackers steal model's training data mostly at inference phase, except the GAN attack [49] and the property inference attack [91] which happen at training phase of collaborative learning.
- ***Countermeasures:*** Concerning the ML pipeline, the protection methods will be applied in data preprocessing phase, training phase, and inference phase, respectively.
- Differential privacy is the most common countermeasure to defend against the stealing attack, however, it alone cannot prevent the GAN attack. Privacy, regularization, dropout, and rounding techniques are popular protections at the training and inference phases.

III. ML-based Stealing Attacks and Protections (Cont.)

3.2.3 Summary (Cont.)

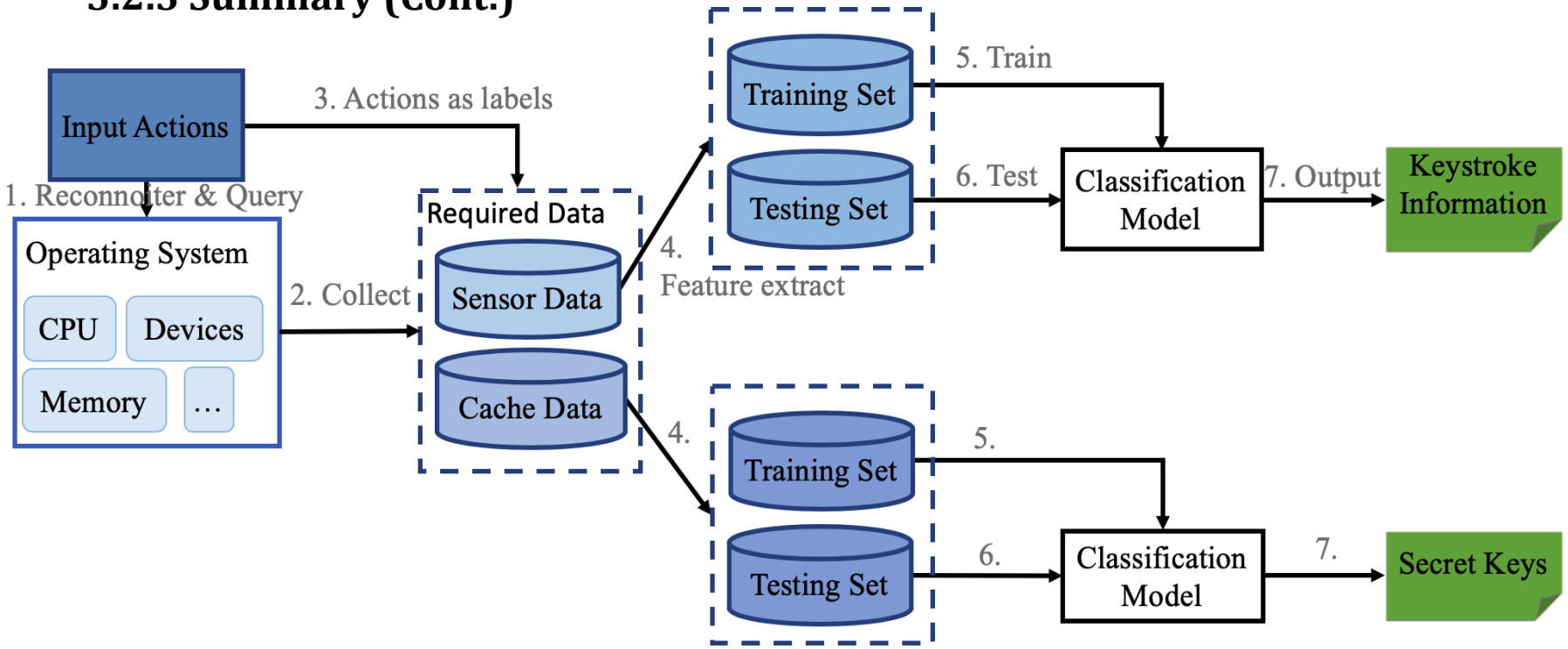


Fig. 6. The ML-based stealing attack against authentication information

- Keystroke information and secret keys. After reconnoitering and querying, attackers targeting at keystroke information and secret keys interact with the target system to collect data, which refers to the active collection.

- The attack involved active collection shares a similar workflow as Fig. 4 depicted.

Thank You

For your Attention!

연락처: yotxaysangthong@seoultech.ac.kr
+82 10-8999-3151

SEOULTECH
SEOUL NATIONAL UNIVERSITY OF SCIENCE & TECHNOLOGY