SoK: The Threat of Offensive AI to Organizations

***Abstract*—AI has provided us with the ability to automate tasks, extract information from vast amounts of data, and synthesize media that is nearly indistinguishable from the real thing. However, positive tools can also be used for negative purposes. In particular, cyber adversaries can use AI to enhance their attacks and expand their campaigns.**

Although offensive AI has been discussed in the past, there is a need to analyze and understand the threat in the context of organizations. For example, how does an AI-capable adversary impact the cyber kill chain? Does AI benefit the attacker more than the defender? What are the most significant AI threats facing organizations today and what will be their impact on the future?

In this SoK, we explore the threat of offensive AI on organizations. First, we present the background and discuss how AI changes the adversary's methods, strategies, goals, and overall attack model. Then, through a literature review, we identify 33 offensive AI capabilities which adversaries can use to enhance their attacks. Finally, through a user study spanning industry and academia, we rank the AI threats and provide insights on the adversaries.

***Index Terms*—Offensive AI, APT, organization security, adversarial machine learning, deepfake, AI-capable adversary**

I. Introduction

For decades, organizations, including government agencies, hospitals, and financial institutions, have been the target of cyber attacks [1]–[3]. These cyber attacks have been carried out by experienced hackers that has involved manual effort. In recent years there has been a boom in the development of artificial intelligence (AI), which has enabled the creation of software tools that have helped to automate tasks such as prediction, information retrieval, and media synthesis. Throughout this period, members of academia and industry have utilized AI[1] in the context of improving the state of cyber defense [4]–[6] and threat analysis [7]–[9]. However, AI is a double edged sword, and attackers can utilize it to improve their malicious campaigns.

Recently, there has been a lot of work done to identify and mitigate attacks on AI-based systems (adversarial machine learning) [10]–[15]. However, an AI-capable adversary can do much more than poison or fool a machine learning model. Adversaries can improve their tactics to launch attacks that were not possible before. For example, with deep learning one can perform highly effective spear phishing attacks by impersonating a superior's face and voice [16], [17]. It is also possible to improve stealth capabilities by using automation to perform lateral movement through a network, limiting command and control (C&C) communication [18], [19]. Other capabilities include the use of AI to find zero-day vulnerabilities in software, automate reverse engineering, exploit side channels efficiently, build realistic fake personas, and to perform many more malicious activities with improved efficacy (more examples are presented later in section IV).

[1]In this paper, we consider machine learning to be a subset of AI technologies.

A. Goal

In this work, we provide a survey of knowledge (SoK) on offensive AI in the context of enterprise security. The goal of this paper is to help the community (1) better understand the current impact of offensive AI on organizations, (2) prioritize research and development of defensive solutions, and (3) identify trends that may emerge in the near future. This work isn't the first to raise awareness of offensive AI. In [20] the authors warned the community that AI can be used for unethical and criminal purposes with examples taken from various domains. In [21] a workshop was held that attempted to identify the potential top threats of AI in criminology. However, these works relate to the threat of AI on society overall and are not specific to organizations and their networks.

B. Methodology

Our SoK was performed in the following way. First, we reviewed literature to identify and organize the potential threats of AI to organizations. Then, we surveyed experts from academia, industry, and government to understand which of these threats are actual concerns and why. Finally, using our survey responses, we ranked these threats to gain insights and to help identify the areas which require further attention. The survey participants were from a wide profile of organizations such as MITRE, IBM, Microsoft, Airbus, Bosch, Fujitsu, Hitachi, and Huawei.

To perform our literature review, we used the MITRE ATT&CK[2] matrix as a guide. This matrix lists the common tactics (or attack steps) which an adversary performs when attacking an organization, from planning and reconnaissance leading to the final goal of exploitation. We divided the tactics among five different academic workgroups from different international institutions based on expertise. For each tactic in the MITRE ATT&CK matrix, a workgroup surveyed related works to see how AI has and can be used by an attacker to improve their tactics and techniques. Finally, each workgroup cross inspected each other's content to ensure correctness and completeness.

C. Main Findings

From the Literature Survey.

- There are three primary motivations for an adversary to use AI: coverage, speed, and success.

[2]https://attack.mitre.org/matrices/enterprise/

- AI introduces new threats to organizations. A few examples include the poisoning of machine learning models, theft of credentials through side channel analysis, and the targeting of proprietary training datasets.
- Adversaries can employ 33 offensive AI capabilities against organizations. These are categorized into seven groups: (1) automation, (2) campaign resilience, (3) credential theft, (4) exploit development, (5) information gathering, (6) social engineering, and (7) stealth.
- Defense solutions, such as AI methods for vulnerability detection [22], pen-testing [23], and credential leakage detection [24] can be weaponized by adversaries for malicious purposes.

From the User Study.

- The top three most threatening categories of offensive AI capabilities against organizations are (1) exploit development, (2) social engineering, and (3) information gathering.
- 24 of the 33 offensive AI capabilities pose significant threats to organizations.
- For the most part, industry and academia are not aligned on the top threats of offensive AI against organizations. Industry is most concerned with AI being used for reverse engineering, with a focus on the loss of intellectual property. Academics, on the other hand, are most concerned about AI being used to perform biometric spoofing (e.g., evading fingerprint and facial recognition).
- Both industry and academia ranked the threat of using AI for impersonation (e.g., real-time deepfakes to perpetrate phishing and other social engineering attacks) as their second highest threat. Jointly, industry and academia feel that impersonation is the biggest threat of all.
- Evasion of intrusion detection systems (e.g., with adversarial machine learning) is considered to be the least threatening capability of the 24 significant threats, likely due to the adversary's inaccessibility to training data.
- AI impacts the cyber kill chain the most during the initial attack steps. This is because the adversary has access to the environment for training and testing of their AI models.
- Because of an AI's ability to automate processes, adversaries may shift from having a few slow covert campaigns to having numerous fast-paced campaigns to overwhelm defenders and increase their chances of success.

D. Contributions

In this SoK, we make the following contributions:

- An overview of how AI can be used to attack organizations and its influence on the cyber kill chain (section III).
- An enumeration and description of the 33 offensive AI capabilities which threaten organizations, based on literature and current events (section IV).
- A threat ranking and insights on how offensive AI impacts organizations, based on a user study with members from academia, industry, and government (section V).
- A forecast of the AI threat horizon and the resulting shifts in attack strategies (section VI).

II. Background on Offensive AI

AI is intelligence demonstrated by a machine. It is often associated as a tool for automating some task which requires some level of intelligence. Early AI models were rule based systems designed using an expert's knowledge [25], followed by search algorithms for selecting optimal decisions (e.g., finding paths or playing games [26]). Today, the most popular type of AI is machine learning (ML) where the machine can gain its intelligence by learning from examples. Deep learning (DL) is a type of ML where an extensive artificial neural network is used as the predictive model. Breakthroughs in DL have led to its ubiquity in applications such as automation, forecasting, and planning due to its ability to reason upon and generate complex data.

A. Training and Execution

In general, a machine learning model can be trained on data with an explicit ground-truth (supervised), with no ground-truth (unsupervised), or with a mix of both (semi-supervised). The trade-off between supervised and non-supervised approaches is that supervised methods often have much better performance at a given task, but require labeled data which can be expensive or impractical to collect. Moreover, unsupervised techniques are open-world, meaning that they can identify novel patterns that may have been overlooked. Another training method is reinforcement learning where a model is trained based on reward for good performance. Lastly, for generating content, a popular framework is adversarial learning. This was first popularised in [27] where the generative adversarial network (GAN) was proposed. A GAN uses a discriminator model to 'help' a generator model produce realistic content by giving feedback on how the content fits a target distribution.

Where a model is trained or executed depends on the attacker's task and strategy. For example, the training and execution of models for reconnaissance tasks will likely take place offsite from the organization. However, the training and execution of models for attacks may take place onsite, offsite, or both. Another possibility is where the adversary uses few-shot learning [28] by training on general data offsite and then fine tuning on the target data onsite. In all cases, the adversary will first design and evaluate their model offsite prior to its usage on the organization to ensure its success and to avoid detection.

For onsite execution, an attacker runs the risk of detection if the model is complex (e.g. a DL model). For example when the model is transferred over to the organization's network or when the attacker's model begins to utilize resources, it may trigger the organization's anomaly detection system. To mitigate this issue, the adversary must consider a trade-off between stealth and effectiveness. For example the adversary may (1) execute the model during off hours or on non-essential devices, (2) leverage an insider to transfer the model, or (3) transfer the observations off-site for execution.

There are two forms of offensive AI: Attacks using AI and attacks against AI. For example, an adversary can (1) use

TABLE I
EXAMPLES OF WHERE A MODEL CAN BE TRAINED AND EXECUTED IN AN ATTACK ON AN ORGANIZATION

Training		Execution		Example
Offsite	Onsite	Offsite	Onsite	
•		•		Vulnerability detection
•			•	Side channel keylogging
	•	•		Channel compression for exfiltration
	•		•	Traffic shaping for evasion
•	•		•	Few-shot learning for record tampering

**Onsite refers to being within the premisis or network of the organization*

AI to improve the efficiency of an attack (e.g., information gathering, attack automation, and vulnerability discovery) or (2) use knowledge of AI to exploit the defender's AI products and solutions (e.g., to evade a defense or to plant a trojan in a product). The latter form of offensive AI is commonly referred to as adversarial machine learning.

B. Attacks Using AI

Although there are a wide variety of AI tasks which can be used in attacks, we found the following to be the most common:

Prediction This is the task of making a prediction based on previously observed data. Common examples are classification, anomaly detection, and regression. Examples of prediction for an offensive purpose includes the identification of keystrokes on a smartphone based on motion [29]–[31], the selection of the weakest link in the chain to attack [32], and the localization of software vulnerabilities for exploitation [22], [33], [34].

Generation This is the task of creating content that fits a target distribution which, in some cases, requires realism in the eyes of a human. Examples of generation for offensive uses include the tampering of media evidence [35], [36], intelligent password guessing [37], [38], and traffic shaping to avoid detection [39], [40]. Deepfakes are another instance of offensive AI in this category. A deepfake is a believable media created by a DL model. The technology can be used to impersonate a victim by puppeting their voice or face to perpetrate a phishing attack [16].

Analysis This is the task of mining or extracting useful insights from data or a model. Some examples of analysis for offense are the use of explainable AI techniques [41] to identify how to better hide artifacts (e.g., in malware) and the clustering or embedding of information on an organization to identify assets or targets for social engineering.

Retrieval This is the task of finding content that matches or that is semantically similar to to a given query. For example, in offense, retrieval algorithms can be used to track an object or an individual in a compromised surveillance system [42], [43], to find a disgruntled employee (as a potential insider) using semantic analysis on social media posts, and to summarize lengthy documents [44] during open source intelligence (OSINT) gathering in the reconnaissance phase.

Decision Making The task of producing a strategic plan or coordinating an operation. Examples of this in offensive AI are the use of swarm intelligence to operate an autonomous botnet [45] and the use of heuristic attack graphs to plan optimal attacks on networks [46].

C. Attacks Against AI - Adversarial Machine Learning

An attacker can use its AI knowledge to exploit ML model vulnerabilities violating its confidentiality, integrity, or availability. Attacks can be staged at either training (development) or test time (deployment) through one of the following attack vectors:

Modify the Training Data. Here the attacker modifies the training data to harm the integrity or availability of the model. Denial of service (DoS) poisoning attacks [47]–[49] are when the attacker decreases the model's performance until it is unusable. A backdoor poisoning attack [50], [51] or trojaning attack [52], is where the attacker teaches the model to recognize an unusual pattern that triggers a behavior (e.g., classify a sample as safe). A triggerless version of this attack causes the model to misclassify a test sample without adding a trigger pattern to the sample itself [53], [54]

Modify the Test Data. In this case, the attacker modifies test samples to have them misclassified [55]–[57]. For example, altering the letters of a malicious email to have it misclassified as legitimate, or changing a few pixels in an image to evade facial recognition [58]. Therefore, these types of attacks are often referred to as evasion attacks. By modifying test samples ad-hoc to increase the model's resource consumption, the attacker can also slow down the model performances. [59].

Analyze the Model's Responses. Here, the attacker sends a number of crafted queries to the model and observes the responses to infer information about the model's parameters or training data. To learn about the training data, there are membership inference [60], deanonymization [61], and model inversion [62] attacks. For learning about the model's parameters there are model stealing/extraction [63], [64], and blind-spot detection [65], state prediction [66].

Modify the Training Code. This is where the attacker performs a supply chain attack by modifying a library used to train ML models (e.g., via an open source project). For example, a compromised loss (training) function that inserts a backdoor [67].

Modify the Model's Parameters. In this attack vector, the attacker accesses a trained model (e.g., via a model zoo or security breach) and tamper its parameters to insert a latent behavior. These attacks can be performed at the software [68], [69], [69] or hardware [70] levels (a.k.a. fault attacks).

Depending on the scenario, an attacker may not have full knowledge or access to the target model:

- **White-Box (Perfect-Knowledge) Attacks:** The attacker knows everything about the target system. This is the

worst case for the system defender. Although it is not very likely to happen in practice, this setting is interesting as it provides an empirical upper bound on the attacker's performance.

- **Gray-Box (Limited-Knowledge) Attacks:** The attacker has partial knowledge of the target system (e.g., the learning algorithm, architecture, etc.) but no knowledge of training data or the model's parameters.
- **Black-Box (Zero-Knowledge) Attacks:** The attacker knows only the task the model is designed to perform and which kind of features are used by the system in general (e.g., if a malware detector has been trained to perform static or dynamic analysis). The attacker may also be able to analyse the model's responses in a black-box manner to get feedback on certain inputs.

In a black or gray box scenario, the attacker can build a surrogate ML model and try to devise the attacks against it as the attacks often transfer between different models. [55], [71].

An attacker does not need to be an expert at machine learning to implement these attacks. Many can be acquired from open-source libraries online [72]–[75].

III. Offensive AI vs Organizations

In this section, we provide an overview of offensive AI in the context of organizations. First we review a popular attack model for enterprise. Then we will identify how an AI-capable adversary impacts this model by discussing the adversary's new motivations, goals, capabilities, and requirements. Later in section IV, we will detail the adversary's techniques based on our literature review.

A. The Attack Model

There are a variety of threat agents which target organizations. These agents are cyber terrorists, cyber criminals, employees, hacktivists, nation states, online social hackers, script kiddies, and other organizations like competitors. There are also some non-target specific agents, such as certain botnets and worms, which threaten the security of an organization. A threat agent may be motivated for various reasons. For example, to (1) make money through theft or ransom, (2) gain information through espionage, (3) cause physical or psychological damage for sabotage, terrorism, fame, or revenge, (4) reach another organization, and (5) obtain foothold on the organization as an asset for later use [76]. These agents not only pose a threat to the organization, but also its employees, customers, and the general public as well (e.g., attacks on critical infrastructure).

In an attack, there may be number of attack steps which the threat agent must accomplish. These steps depend on the adversary's goal and strategy. For example, in an advanced persistent threat (APT) [77]–[79], the adversary may need to reach an asset deep within the defender's network. This would require multiple steps involving reconnaissance, intrusion, lateral movement through the network, and so on. However, some attacks can involve just a single step. For example, a spear phishing attack in which the victim unwittingly provides confidential information or even transfers money. In this paper, we describe the adversary's attack steps using the MITRE ATT&CK Matrix for Enterprise[3] which captures common adversarial tactics based on real-world observations.

Attacks which involve multiple steps can be thwarted if the defender identifies or blocks the attack early on. The more progress which an adversary makes, the harder it is for the defender to mitigate it. For example, it is better to stop a campaign during the initial intrusion phase than during the lateral movement phase where an unknown number of devices in the network have been compromised. This concept is referred to as the *cyber kill chain*. From an offensive perspective, the adversary will want shorten and obscure the kill chain by being as to be as efficient and covert as possible. In particular, operation within a defender's network usually requires the attacker to operate through a remote connection or send commands to compromised devices (bots) from a command and control (C2). This generates presence in the defenders network which can be detected over time.

B. The Impact of Offensive AI

Conventional adversaries use manual effort, common tools, and expert knowledge to reach their goals. In contrast, an AI-capable adversary can use AI to automate its tasks, enhance its tools, and evade detection. These new abilities affect the cyber kill chain.

First, let's discuss why an adversary would consider using AI in its offensive on an organization.

1) The Three Motivators of Offensive AI: In our survey, we found that there are three core motivations for an adversary to use AI in an offensive against an organization: coverage, speed, and success.

Coverage. By using AI, an adversary can scale up its operations through automation to decrease human labor and increase the chances of success. For example, AI can be used to automatically craft and launch spear phishing attacks, distil and reason upon data collected from OSINT, maintain attacks on multiple organizations in parallel, and reach more assets within a network to gain a stronger foothold. In other words, AI enables adversaries to target more organizations with higher precision attacks with a smaller workforce.

Speed. With AI, an adversary can reach its goals faster. For example, machine learning can be used to help extract credentials, intelligently select the next best target during lateral movement, spy on users to obtain information (e.g., perform speech to text on eavesdropped audio), or find zero-days in software. By reaching a goal faster, the adversary not only saves time for other ventures but can also minimize its presence (duration) within the defender's network.

Success. By enhancing its operations with AI, an adversary increases its likelihood of success. Namely, ML can be used to (1) make the operation more covert by minimizing

[3]https://attack.mitre.org/

or camouflaging network traffic (such as C2 traffic) and by exploiting weaknesses in the defender's AI models such as an ML-based intrusion detection system (IDS), (2) identify opportunities such as good targets for social engineering attacks and novel vulnerabilities, (3) enable better attack vectors such as using deepfakes in spear phishing attacks, (4) plan optimal attack strategies, and (5) strengthen persistence in the network through automated bot coordination and malware obfuscation.

We note that these motivations are not mutually exclusive. For example, the use of AI to automate a phishing campaign increases coverage, speed, and success.

2) AI-Capable Threat Agents: It is clear that some AI-capable threat agents will be able to perform more sophisticated AI attacks than others. For example, state actors can potentially launch intelligent automated botnets where hacktivists will likely struggle in accomplishing the same. However, we have observed over the years that AI has become increasingly accessible, even to novice users. For example, there are a wide variety of open source deepfakes technologies online which are plug and play[4]. Therefore, the sophistication gap between certain threat agents may close over time as the availability to AI technology increases.

3) New Attack Goals: In addition to the conventional attack goals, AI-capable adversaries have new attack goals as well:

Sabotage. The adversary may want to use its knowledge of AI to cause damage to the organization. For example, it may want to alter ML models in the organization's products and solutions by poisoning their dataset to alter performance or by planting a trojan in the model for later exploitation. Moreover, the adversary may want to perform an adversarial machine learning attack on an AI system. For example, to evade detection in surveillance [58] or to tip financial or energy forecasts models in the adversary's favor. Finally, the adversary may also use generative AI to add or modify evidence in a realistic manner. For example, to modify or plant evidence in surveillance footage [80], medical scans [35], or financial records [36].

Espionage. With AI, an adversary can improve its ability to spy on organizations and extract/infer meaningful information. For example, they can use speech to text algorithms and sentiment analysis to mine useful audio recordings [81] or steal credentials through acoustic or motion side channels [82], [83]. AI can also be used to extract latent information from encrypted web traffic [84], and track users through the organization's social media [85]. Finally, the attacker may want to achieve an autonomous persistent foothold using swarm intelligence [18].

Information Theft. An AI-capable adversary may want to steal models trained by the organization to use in future white box adversarial machine learning attacks. Therefore, some data records and proprietary datasets may be targeted for the sake of training models. In particular, audio or video records of customers and employees may be stolen to create convincing deepfake impersonations. Finally, intellectual property may be targeted through AI powered reverse engineering tools [86].

[4]https://github.com/datamllab/awesome-deepfakes-materials

4) New Attack Capabilities: Through our survey, we have identified 33 offensive AI capabilities (OAC) which directly improve the adversary's ability to achieve attack steps. These OACs can be grouped into seven OAC categories: (1) automation, (2) campaign resilience, (3) credential theft, (4) exploit development, (5) information gathering, (6) social engineering, and (7) stealth. Each of these capabilities can be tied to the three motivators introduced in section III-B1.

In Fig. 1, we present the OACs and map their influence on the cyber kill chain (the MITRE enterprise ATT&CK model). An edge in the figure means that the indicated OAC improves attacker's ability to achieve the given attack step. From the figure, we can see that offensive AI impacts every aspect of the attack model. Later in section IV we will discuss each of these 33 OACs in greater detail.

These capabilities are materialized in one of two ways:

AI-based tools are programs which performs a specific task in adversary's arsenal. For example, a tool for intelligently predicting passwords [37], [38], obfuscating malware code [87], traffic shaping for evasion [39], [40], [88], puppeting a persona [16], and so on. These tools are typically in the form of a machine learning model.

AI-driven bots are autonomous bots which can perform one or more attack steps without human intervention, or coordinate with other bots to efficiently reach their goal. These bots may use a combination of swarm intelligence [45] and machine learning to operate.

IV. Survey of Offensive AI Capabilities

In section III-B4 we presented the 33 offensive AI capabilities. We will now describe each of the OACs in order of their 7 categories: automation, campaign resilience, credential theft, exploit development, information gathering, social engineering, and stealth.

A. Automation

The process of automation gives adversaries a hands-off approach to accomplishing attack steps. This not only reduces effort, but also increases the adversary's flexibility and enables larger campaigns which are less dependent on C2 signals.

1) Attack Adaptation: Adversaries can use AI to help adapt their malware and attack efforts to unknown environments and find their intended targets. For example, identifying a system [89] before attempting an exploit to increase the chances of success and avoid detection. In Black Hat'18, IBM researchers showed how a malware can trigger itself using DL by identifying a target's machine by analysing the victim's face, voice, and other attributes. With models such as decision trees, malware can locate and identify assets via complex rules like [90], [91]. Instead of transferring screenshots [92]–[95] DL can be used onsite to extract critical information.

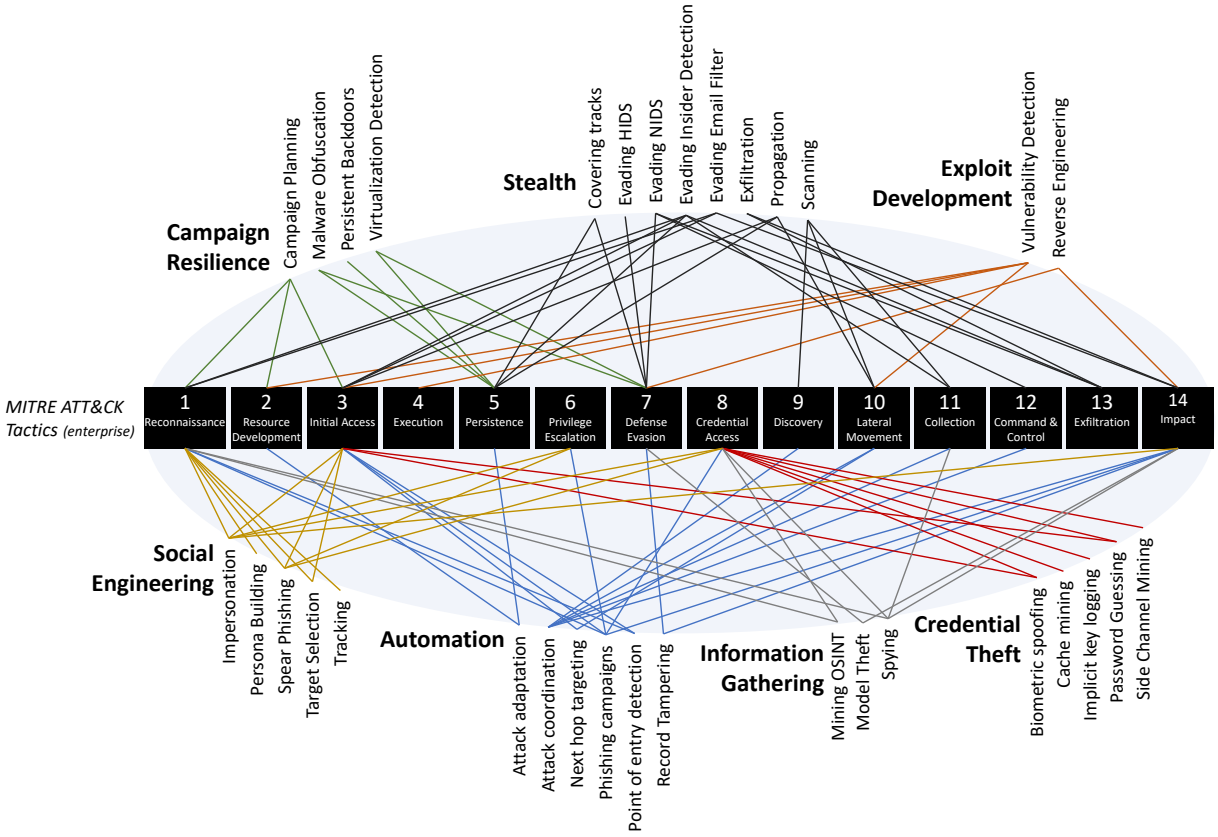

Fig. 1. The 33 offensive AI capabilities (OAC) identified in our survey, mapped to the MITRE enterprise ATT&CK model. An edge indicates that the OAC directly helps the attacker achieve the indicated attack step.

2) Attack Coordination: Cooperative bots can use AI to find the best times and targets to attack. For example, swarm intelligence [96] is the study of autonomous coordination among bots in a decentralized manner. Researchers have proposed that botnets can use swarm intelligence as well. In [18] the authors discuss a hypothetical swarm malware and in [19] the authors propose another which uses DL to trigger attacks. AI bots can also communicate information on asset locations to fulfill attacks (e.g., send a stolen credential or relevant exploit to a compromised machine).

3) Next hop targeting: During lateral movement, the adversary must select the next asset to scan or attack. Choosing poorly may prolong the attack and risk detection by the defenders. For example, consider a browser like Firefox which has 4325 key-value pairs denoting the individual configurations. Only some inter-plays of these configurations are vulnerable [97], [98]. Reinforcement learning can be used to train a detection model which can identify the best browser to target. As for planning multiple steps, a strategy can be formed by using reinforcement learning on Petri nets [46] where attackers and defenders are modeled as competing players. Another approach is to use DL [99], [100] to explore "attack graphs" [101] that contain the target's network structure and the vulnerabilities. Notably, the Q-learning algorithms have enabled the approach to work on large-scale enterprise networks [102].

4) Phishing Campaigns: Phishing campaigns involve sending the same emails or robo-phone calls in mass. When someone falls prey and responds, the adversary takes over the conversation. These campaigns can be fully automated through AI like Google's assistant which can make phone calls on your behalf [103]–[105]. Furthermore, adversaries can increase their success through mass spear phishing campaigns powered with deepfakes, where (1) a bot calls a colleague of the victim (found via social media), (2) clones his/her voice with 5 seconds of audio [106], and then (3) calls the victim in the colleague's voice to exploit their trust.

5) Point of Entry Detection: The adversary can use AI to identify and select the best attack vector for an initial infection. For example, in [107] statistical models on an organization's attributes were used to predict the number of intrusions it receives. The adversary can train a model on similar information to select the weakest organizations (low hanging fruits) and the strongest attack vectors.

6) Record Tampering: An adversary may use AI to tamper records as part of their end-goal. For example, ML can be used to impact business decisions with synthetic data [108], to obstruct justice by tampering evidence [80], to perform fraud [36] or to modify medical or satellite imagery [35]. As shown in [35], DL-tampered records can fool human observers and can be accomplished autonomously onsite.

B. Campaign Resilience

In a campaign, adversaries try to ensure that their infrastructure and tools have a long life. Doing so helps maintain a foothold in the organization and enables reuse of tools and exploits for future and parallel campaigns. AI can be used to improve campaign resilience through planning, persistence, and obfuscation.

1) Campaign Planning: Some attacks require careful planning long before the attack campaign to ensure that all of the attacker's tools and resources are obtainable. ML-based cost benefit analysis tools, such as in [109], may be used to identify which tools should be developed and how the attack infrastructure should be laid out (e.g., C2 servers, staging areas, etc). It could also be used to help identify other organizations that can be used as beach heads [76]. Moreover, ML can be used to plan a digital twin [110], [111] of the victim's network (based on information from reconnaissance) to be created offsite for tuning AI models and developing malware.

2) Malware Obfuscation: ML models such as GANs can be used to obscure a malware's intent from an analyst. Doing so can enable reuse of the malware, hide the attacker's intents and infrastructure, and prolong an attack campaign. The concept is to take an existing piece of software and emit another piece that is functionally equivalent (similar to translation in NLP). For example, DeepObfusCode [87] uses recurrent neural networks (RNN) to generate ciphered code. Alternatively, backdoors can be planted in open source projects and hidden using similar manners [112].

3) Persistent Access: An adversary can have bots establish multiple back doors per host and coordinate reinfection efforts among a swarm [18]. Doing so achieves a foothold on an organization by slowing down the effort to purge the campaign. To avoid detection in payloads deployed during boot, the adversary can use a two-step payload which uses ML to identify when to deploy the malware and avoid detection [113], [114]. Moreover, a USB sized neural compute stick[5] can be planted by an insider to enable covert and autonomous onsite DL operations.

4) Virtualization Detection: To avoid dynamic analysis and detection in sandboxes, an adversary may try to have the malware detect the sandbox before triggering. The malware could use ML to detect a virtual environment by measuring system timing (e.g., like in [115]) and other system properties.

C. Credential Theft

Although a system may be secure in terms of access control, side channels can be exploited with ML to obtain a user's credentials and vulnerabilities in AI systems can be used to avoid biometric security.

1) Biometric spoofing: Biometric security is used for access to terminals (such as smartphones) and for performing automated surveillance [116]–[118]. Recent works have shown how AI can generate "Master Prints" which are deepfakes of fingerprints that can open nearly any partial print scanner (such as on a smartphone) [119]. Face recognition systems can be fooled or evaded with the use of adversarial samples. For example, in [58] where the authors generated colorful glasses that alters the perceived identity. Moreover, 'sponge' samples [59] can be used to slow down a surveillance camera until it is unresponsive or out of batteries (when remote). Voice authentication can also be evaded through adversarial samples, spoofed voice [120], and by cloning the target's voice with deep learning [120].

2) Cache mining: Information on credentials can be found in a system's cache and log dumps, but the large amount of data makes finding it a difficult task. However, the authors of [121] showed how ML can be used to identify credentials in cache dumps from graphic libraries. Another example is the work of [24] where an ML system was used to identify cookies containing session information.

3) Implicit key logging: Over the last few years researchers have shown how AI can be used as an implicit key-logger by sensing side channel information from a physical environment. The side channels comes in one or a combination of the following aspects:

Motion. When tapping on a phone screen or typing on a keyboard, the device and nearby surfaces move and vibrate. A malware can use the smartphone's motion sensors to decipher the touch strokes on the phone [29], [30] and keystrokes on nearby keyboards [31]. Wearable devices can be exploited in a similar way as well [122], [123].

Audio. Researchers have shown that, when pressed, each key gives of it's own unique sound which can be used to infer what is being typed [82], [124]. Timing between key strokes is also a revealing factor due to the structure of the language and keyboard layout. Similar approaches have also been shown for inferring touches on smartphones [83], [125], [126].

Video. In some cases, a nearby smartphone or compromised surveillance camera can be used to observe keystrokes, even when the surface is obscured. For example, via eye movements [127]–[129], device motion [130], and hand motion [131], [132].

4) Password Guessing: Humans tend to select passwords with low entropy or with personal information such as dates. GANs can be used to intelligently brute-force passwords by learning from leaked password databases [37]. Researchers have improved on this approach by using RNNs in the generation process [133]. However, the authors of [38] found that models like [37] do not work well on Russian passwords. Instead, adversaries may pass the GAN personal information on the user to improve the performance [134].

5) Side Channel Mining: ML algorithms are adept at extracting latent patterns in noisy data. Adversaries can leverage ML to extract secrets from side channels emitted from cryptographic algorithms. This has been accomplished on a variety of side channels including power consumption [135], [136], electromagnetic emanations [137], processing time [138], cache hits/misses [115]. In general, ML can be used to mine nearly

[5]https://software.intel.com/content/www/us/en/develop/articles/intel-movidius-neural-compute-stick.html

any kind of side channel [139]–[146]. For example, credentials can be extracted from the timing of network traffic [147].

D. Exploit Development

Adversaries work hard to understand the content and inner-workings of compiled software to (1) steal intellectual property, (2) share trade secrets, (3) and identify vulnerabilities which they can exploit.

1) Reverse Engineering: While interpreting compiled code, an adversary can use ML to help identify functions and behaviors, and guide the reversal process. For example binary code similarity can be used to identify well-known or reused behaviors [148]–[154] and autoencoder networks can be used to segment and identify behaviors in code, similar to the work of [7]. Furthermore, DL can potentially be used to lift compiled code up to a higher-level representation using graph transformation networks [155], similar to semantic analysis in language processing. Protocols and state machines can also be reversed using ML. For example, CAN bus data in a vehicles [156], network protocols [157], and commands [158], [159].

2) Vulnerability Detection: There are a wide variety of software vulnerability detection techniques which can be broken down into static and dynamic approaches:

Static. For open source applications and libraries, the attacker can use ML tools for detecting known types of vulnerabilities in source code [34], [160]–[163]. If its a commercial product (compiled as a binary) then methods such as [7] can be used to identify vulnerabilities by comparing parts of the program's control flow graph to known vulnerabilities.

Dynamic. ML can also be used to perform guided input 'fuzzing' which can reach buggy code faster [22], [164]–[169]. Many works have also shown how AI can mitigate the issue of symbolic execution's massive state space [33], [170]–[173].

E. Information Gathering

AI scales well and is very good at data mining and language processing. These capabilities can be used by an adversary to collect and distil actionable intel for a campaign.

1) Mining OSINT: In general, there are three ways in which AI can improve an adversary's OSINT.

Stealth. The adversary can use AI to camouflage its probe traffic to resemble benign services like Google's web crawler [9]. Unlike heavy tools like Metagoofil [174], ML can be used to minimize interactions by prioritizing sites and data elements [175], [176].

Gathering. Network structure and elements can be identified using cluster analysis or graph-based anomaly detection [177]. Credentials and asset information can be found using methods like reinforcement learning on other organizations [178]. Finally, personnel structure can be extracted from social media using NLP-based web scrappers like Oxylabs [179].

Extraction. Techniques like NLP can be used to translate foreign documents [180], identify relevant documents [181], [182], extract relevant information from online sources [183], [184], and locate valid identifiers [85].

2) Model Theft: An adversary may want to steal an AI model to (1) obtain it as intellectual property, (2) extract information about members of its training set [60]–[62], or (3) use it to perform a white-box attack against an organization. As described in section II-C, if the model can be queried (e.g., model as a service -MAAS), then its parameters [63], [64] and hyperparameters [185] can be copied by observing the model's responses. This can also be done through side-channel [186] or hardware-level analysis [187].

3) Spying: DL is extremely good at processing audio and video, and therefore can be used in spyware. For example, a compromised smartphone can map an office by (1) modeling each room with ultrasonic echo responses [188], (2) using object recognition [189] to obtain physical penetration info (control terminals, locks, guards, etc), and (3) automatically mine relevant information from overheard conversations [181], [190]. ML can also be used to analyze encrypted traffic. For example it can extract transcripts from encrypted voice calls [191], identify applications [192], and reveal internet searches [84].

F. Social Engineering

The weakest links in an organization's security are its humans. Adversaries have long targeted humans by exploiting their emotions and trust. AI provides adversaries will enhanced capabilities to exploit humans further.

1) Impersonation (Identity Theft): An adversary may want to impersonate someone for a scam, blackmail attempt, a defamation attack, or to perform a spear phishing attack with their identity. This can be accomplished using deepfake technologies which enable the adversary to reenact (puppet) the voice and face of a victim, or alter existing media content of a victim [16]. Recently, the technology has advanced to the state where reenactment can be performed in real-time [193], and training only requires a few images [194] or seconds of audio [106] from the victim. For high quality deepfakes, large amounts of audio/video data is still needed. However, when put under pressure, a victim may trust a deepfake even if it has a few abnormalities (e.g., in a phone call) [195]. Moreover, the audio/video data may be an end-goal and inside the organization (e.g., customer data).

2) Persona Building: Adversaries build fake personas on online social networks (OSN) to connect with their targets. To evade fake profile detectors, a profile can be cloned and slightly altered using AI [196]–[198] so that they will appear different yet reflect the same personality. The adversary can then use a number of AI techniques to alter or mask the photos from detection [199]–[202]. To build connections, a link prediction model can be used to maximize the acceptance rate [203], [204] and a DL chatbot can be used to maintain the conversations [205].

3) Spear Phishing: Call-based spear phishing attacks can be enhanced using real-time deepfakes of someone the victim trusts. For example, this occured in 2019 when a CEO was

scammed out $240k [17]. For text-based phishing, tweets [23] and emails [134], [206], [207] can be generated to attract a specific victim, or style transfer techniques can be used to mimic a colleague [208], [209].

4) Target Selection: An adversary can use AI to identify victims in the organization who are the most susceptible to social engineering attacks [32]. A regression model based on the target's social attributes (conversations, attended events, etc) can be used as well. Moreover, sentiment analysis can be used to find disgruntled employees to be recruited as insiders [81], [210]–[213].

5) Tracking: To study members of an organization, adversaries may track the member's activities. With ML, an adversary can trace personnel across different social media sites by content [85] and through facial recognition [214]. ML models can also be used on OSN content to track a member's location [215]. Finally, ML can also be used to discover hidden business relationships [216], [217] from the news and from OSNs as well [218], [219].

G. Stealth

In multi step attacks, covert operations are necessary to ensure success. An adversary can either use or abuse AI to evade detection.

1) Covering tracks: To hide traces of the adversary's presence, anomaly detection can be performed on the logs to remove abnormal entries [220], [221]. CryptoNets [222] can also be used to hide malware logs and onsite training data for later use. To avoid detection onsite, trojans can be planted in DL intrusion detection systems (IDS) in a supply chain attack at both the hardware [70], [223] and software [52], [224] levels. DL hardware trojans can use adversarial machine learning to avoid being detected [225].

2) Evading HIDS (Malware Detectors): The struggle between security analysts and malware developers is a never-ending battle, with the malware quickly evolving and defeating detectors. In general, state-of-the-art detectors are vulnerable to evasion [226]–[228]. For example, adversary can evade an ML-based HIDS that performs dynamic analysis by splitting the malware's code into small components executed by different processes [229]. They can also evade ML-based detectors that perform static analysis by adding bytes to the executable [230] or code that does not affect the malware behavior [114], [231]–[234]. Modifying the malware without breaking its malicious functionality is not easy. Attackers may use AI explanation tools like LIME [41] to understand which parts of malware are being recognized by the detector and change them manually. Tools for evading ML-based detection can be found freely online [6].

3) Evading NIDS (Network Intrusion Detection Systems): There are several ways an adversary can use AI to avoid detection while entering, traversing, and communicating over an organization's network. Regarding URL-based NIDSs, attackers can avoid phishing detectors by generating URLS that do not match known examples [235]. Bots trying to contact their C2 server can generate URLs that appear legitimate to humans [236], or that can evade malicious-URL detectors [237]. To evade traffic-based NIDSs, adversaries can shape their traffic [39], [40] or change their timing to hide it [238].

4) Evading Insider Detectors: To avoid insider detection mechanisms, adversaries can mask their operations using ML. For example, given some user's credentials, they can use information on the user's role and the organization's structure to ensure that operation performed looks legitimate [239].

5) Evading Email Filter: Many email services use machine learning to detect malicious emails. However, adversaries can use adversarial machine learning to evade detection [240]–[243]. Similarly, malicious documents attached to emails, containing malware, can evade detection as well (e.g., [244]). Finally, an adversary may send emails to be intentionally detected so that they will be added to the defender's training set, as part of a poisoning attack [245].

6) Exfiltration: Similar to evading NIDSs, adversaries must evade detection when trying to exfiltrate data outside of the network. This can be accomplished by shaping traffic to match the outbound traffic [88] or by encoding the traffic within a permissible channel like Facebook chat [246]. To hide the transfer better, an adversary could use DL to compress [247] and even encrypt [248] the data being exfiltrated. To minimize throughput, audio and video media can be summarized to textual descriptions onsite with ML before exfiltration. Finally, if the network is air gapped (isolated from the Internet) [249] then DL techniques can be used to hide data within side channels such as noise in audio [250].

7) Propagation & Scanning: For stealthy lateral movement, an adversary can configure their Petri nets or attack graphs (see section IV-A3) to avoid assets and subnets with certain IDSs and favour networks with more noise to hide in. Moreover, AI can be used to scan hosts and networks covertly by modeling its search patterns and network traffic according to locally observed patterns [88].

V. User Study & Threat Ranking

In our literature review (section IV) we identified the potential offensive AI capabilities (OAC) which an adversary can use to attack an organization. However, some OACs may be impractical, where others may pose much larger threats. Therefore, we performed a user study to rank these threats and understand their impact on the cyber kill chain.

A. Survey Setup

We surveyed 22 experts in both subjects of AI and cybersecurity. Our participants were CISOs, researchers, ethics experts, company founders, research managers, and other relevant professions. Exactly half of the participants were from academia and the other half were from industry (companies and government agencies). For example, some of our participants were from MITRE, IBM Research, Microsoft, Airbus, Bosch (RBEI), Fujitsu Ltd., Hitachi Ltd., Huawei Technologies, Nord Security, Institute for Infocomm Research

[6] https://github.com/zangobot/secml_malware

(I2R), Purdue University, Georgia Institute of Technology, Munich Research Center, University of Cagliari, and the Nanyang Technological University (NTU). The responses of the participants have been anonymized and reflect their own personal views and not the views of their employers.

The survey consisted of 204 questions which asked the participants to (1) rate different aspects of each OAC, (2) give their opinion on the utility of AI to the adversary in the cyber kill chain, and (3) give their opinion on the balance between the attacker and defender when both have AI. We used these responses to produce threat rankings and to gain insights on the threat of offensive AI to organizations.

Only 22 individuals participated in the survey because AI-cybersecurity experts are very busy and hard to reach. However, assuming there are 100k eligible respondents in the population, with a confidence level of 95% we calculate that we have a margin of error of about 20%. Moreover, since we have sampled a variety of major universities and companies, and since deviation in the responses is relatively small, we believe that the results capture a fair and meaningful view of the subject matter.

B. Threat Ranking

In this section we measure and rank the various threats of an adversary which can utilize or exploit AI technologies to enhance their attacks. For each OAC the participants were asked to rate four aspects on the range of 1-7 (low to high):

Profit (P): The amount of benefit which a threat agent gains by using AI compared to using non-AI methods. For example, attack success, flexibility, coverage, automation, and persistence. Here profit assumes that the AI tool has already been implemented.

Achievability (A): How easy is it for the attacker to use AI for this task considering that the adversary must implement, train, test and deploy the AI.

Defeatability (D): How easy is it for the defender to detect or prevent the AI-based attack. Here, a higher score is bad for the adversary (1=hard to defeat, 7=easy to defeat).

Harm (H): The amount of harm which an AI-capable adversary can inflict in terms of physical, physiological, or monetary damage (including effort put into mitigating the attack).

We say that an adversary is motivated to perform an attack if there is high profit P and high achievability A. Moreover, if there is high P but low A or vice versa, some actors may be tempted to try anyways. Therefore, we model the motivation of using an OAC as $M = \frac{1}{2}(P + A)$. However, just because there is motivation, it does not mean that there is a risk. If the AI attack can be easily detected or prevented, then no amount of motivation will make the OAC a risk. Therefore, we model risk as $R = \frac{M}{D}$ where a low defeatability (hard to prevent) increases R and a high defeatability (easy to prevent) lowers R. Risk can also be viewed as the likelihood of the attack occurring, or the likelihood of an attack success. Finally, to model threat, we must consider the amount of harm done to the organization. An OAC with high R but no consequences is less of a threat. Therefore, we model our threat score as

$$T = H\frac{\frac{1}{2}(P + A)}{D} = H\frac{M}{D} = HR \tag{1}$$

Before computing T, we normalize P, A, D, and H from the range 1-7 to 0-1. This way, a threat score greater than 1 indicates a significant threat because for these scores (1) the adversary will attempt the attack ($M > D$), and (2) the level of harm will be greater than the ability to prevent the attack ($\frac{D}{M} < H \leq 1$). We can also see from our model that as an adversary's motivation increases over defeatability, the amount of harm deemed threatening decreases. This is intuitive because if an attack is easy to achieve and highly profitable, then it will be performed more often. Therefore, even if it is less harmful, attacks will occur frequently so the damage will be higher in the long run.

1) OAC Threat Ranking: In Fig. 2 we present the average P, A, D, and H scores for each OAC. In Fig. 3 we present the OACs ranked according to their threat score T, and contrast their risk scores R to their harm scores H.

The results show that 23 of the OACs (72%) are considered to be significant threats (have a $T > 1$). In general we observe that the top threats mostly relate to social engineering and malware development. The top three OACs are impersonation, spear phishing, and model theft. These OACs have significantly larger threat scores than the others because they are (1) easy to achieve, (2) have high payoffs, (3) are hard to prevent, and (4) cause the most harm (top left of Fig. 2). Interestingly, the use of AI to run phishing campaigns is considered a large threat even though it has a relatively high D score. We believe this is because, with AI, an adversary can both increase the number and quality of the phishing attacks. Therefore, even if 99% of the attempts fail, some will get through and cause the organization damage. The least significant threats were scanning and cache mining which are perceived to have have little benefit for the adversary because they pose a high risk of detection. Other low ranked threats include some on-site automation for propagation, target selection, lateral movement, and covering tracks.

2) Industry vs Academia: In Fig. 4 we look at the average threat scores for each OAC *category*, and contrast the opinions of members from academia to those from industry.

In general, academia views AI as a more significant threat to organizations than industry. One can argue that the discrepancy is because industry tends to be more practical and grounded in the present, where academia considers potential threats thus considering the future. For example, when looking at the threat scores from academia, all of the categories are considered significant threats ($T > 1$). However, when looking at the industry's responses, the categories of stealth, credential theft, and campaign resilience are not. This may be because these concepts have presented (proven) themselves less in the wild than the others.

Regardless, both industry and academia agree on the top three most threatening OAC categories: (1) exploit development, (2) social engineering, and (3) information

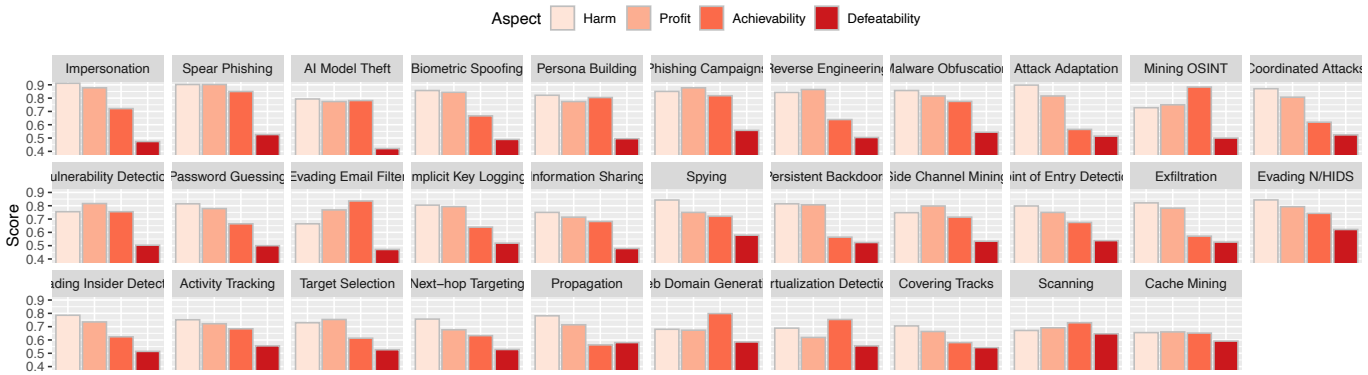

Fig. 2. Survey results: the averaged and normalized opinion scores for each offensive AI capability (OAC) when used against an organization. The OACs are ordered according to their threat score, left to right starting from the first row.

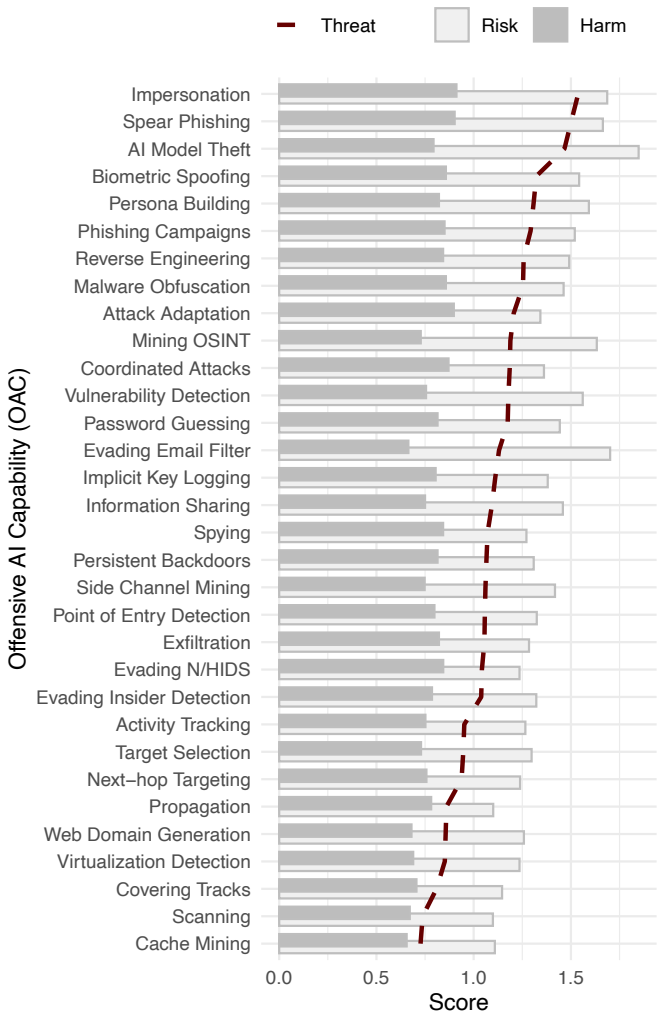

Fig. 3. Survey results: the offensive AI capabilities ranked according to their threat scores.

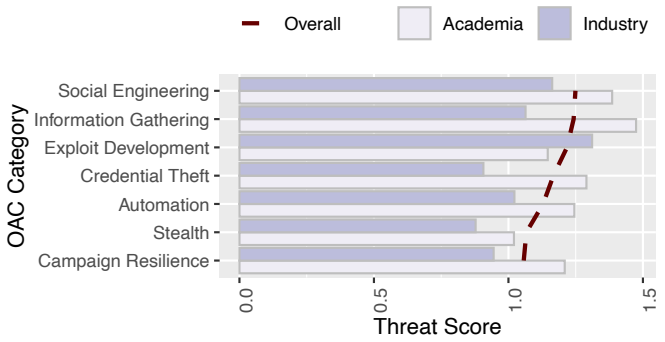

Fig. 4. Survey results: the offensive AI capability categories ranked according to their average threat scores. The scores from industry and academia participants are also presented separately.

gathering. This is because, for these categories, the attacker benefits greatly from using AI (P), can easy implement the relevant AI tools (A), the attack causes considerable damage (H), and there is little the defender can do to prevent them (D) (indicated in Fig. 2). For example, deepfakes are easy to implement yet hard to detect in practice (e.g., in a phone call), and extracting private information from side channels and online resources can be accomplished with little intervention.

Surprisingly, both academia and industry consider the use of AI for stealth as the least threatening OAC category in general. Even though there has been a great deal of work showing how IDS models are vulnerable [39], [230], IDS evasion approaches were considered the second most defeatable OAC after intelligent scanning. This may have to do with the fact that the adversary cannot evaluate its AI-based evasion techniques inside the actual network, and thus risks detection.

Overall, there were some disagreements between industry and academia regarding the most threatening OACs. The top-10 most threatening OACs for organizations (out of 33) were ranked as follows:

Industry's Perspective	Academia's Perspective
1) Reverse Engineering	1) Biometric Spoofing
2) Impersonation	2) Impersonation
3) AI Model Theft	3) Spear Phishing
4) Spear Phishing	4) AI Model Theft
5) Persona Building	5) Mining OSINT
6) Phishing Campaigns	6) Spying
7) Information Sharing	7) Target Selection
8) Malware Obfuscation	8) Side Channel Mining
9) Vulnerability Detection	9) Coordinated Attacks
10) Password Guessing	10) Attack Adaptation

We note that academia views biometric spoofing as the top threat, where industry doesn't consider it in their top 10. We think this is because the latest research on this topic involves ML which can be evaded (e.g., [58], [119]). In contrast to academia, industry views this OAC as less harmful to the organization and less profitable to the adversary, perhaps because biometric security is not a common defense used in organization. Regardless, biometric spoofing is still considered the 4-th highest threat overall (Fig. 3). Another insight is that academia is more concerned about the use of ML for spyware, side-channels, target selection, and attack adaptation than industry. This may be because these are topics which have long been discussed in academia, but have yet to cause major disruptions in the real-world. For industry, they are more concerned with the use of AI for exploit development and social engineering, likely because these are threats which are out of their control.

Additional figures which compare the responses of industry to academia can be found online[7].

C. Impact on the Cyber Kill Chain

For each of the 14 MITRE ATT&CK steps, we asked the participants whether they agree or disagree[8] to the following statements: (1) It more beneficial for the attacker to use AI than conventional methods in this attack step, and (2) AI benefits the attacker more than AI benefits the defender. The objective of these questions were to identify how AI impacts the kill chain and whether AI forms any asymmetry between the attacker and defender.

In Fig. 5 we present the mean opinion scores along with their standard deviations (additional histograms can be found online[7]). Overall, our participants felt that AI enhances the adversary's ability to traverse the kill chain. In particular, we observe that adversary benefits considerably from AI during the first three steps. One explanation is that these attacks are maintained offsite and thus are easier to develop and have less risk. Moreover, we understand from the results that there is a general feeling that defenders do not have a good way to preventing adversarial machine learning attacks. Therefore, AI not only improves defense evasion but also gives the attacker a considerable advantage over the defender in this regard.

Our participants also felt that an adversary with AI has a somewhat greater advantage over a defender with AI for most

[7]https://tinyurl.com/t735m6st

[8]Measured using a 7-step likert scale ranging from strongly disagree (-3) to neutral (0) to strongly agree (+3).

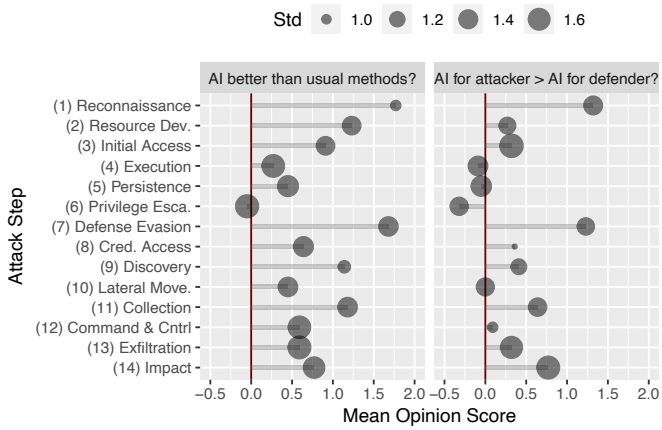

Fig. 5. Survey results: Mean opinion scores on whether (1) it is more beneficial for the adversary to use AI over conventional methods, and (2) AI benefits attackers more than AI benefits defenders. The scores range from -3 to +3.

attack steps. In particular, the defender cannot effectively utilize AI to prevent reconnaissance except for mitigating a few kinds of social engineering attacks. Moreover, the adversary has many new uses for AI during the impact step, such as the tampering of records, where the defender does not. However, the participants felt that the defender has an advantage when using AI to detect execution, persistence, and privilege escalation. This is understandable since the defender can train and evaluate models onsite whereas the attacker cannot.

VI. Discussion

In this section, we share our insights on our findings and discuss the road ahead.

A. Insights, Observations, & Limitations

Top Threats. It is understandable why the highest ranked threats to organizations relate to social engineering attacks and software analysis (vulnerability detection and reverse engineering). It is because these attacks are out of the defender's control. For example, humans are the weakest link, even with security awareness training. However, with deepfakes, even less can be done to mitigate these social engineering attacks. The same holds for software analysis where ML has proven itself to work well with languages and even compiled binaries [154]. As mentioned earlier, we believe the reason academia is the most concerned with biometrics is because it almost exclusively uses ML, and academia is well aware of ML's flaws. On the other hand, industry members know that organizations do not often employ biometric security. Therefore, they perceive AI attacks on their software and personnel as the greatest threats.

The Near Future. Over the next few years, we believe that there will be an increase of offensive AI incidents, but only at the front and back of the attack model (recon., resource development, and impact –such as record tampering). This is because currently AI cannot effectively learn on its own. Therefore, we aren't likely to see botnets that can autonomously and dynamically interact with a diverse set of complex systems

(like an organization's network) in the near future. Therefore, since modern adversaries have limited information on the organizations' network, they are restricted to attacks where the data collection, model development, training, and evaluation occur offsite. In particular, we note that DL models are large and require a considerable amount of resources to run. This makes them easy to detect when transferred into the network or executed onsite. However, the model's footprint will become less anomalous over time as DL proliferates. In the near future, we also expect that phishing campaigns will become more rampant and dangerous as humans and bots are given the ability to make convincing deepfake phishing calls.

AI is a Double Edged Sword. We observed that AI technologies for security can also be used in an offensive manner. Some technologies are dual purpose. For example, the ML research into disassembly, vulnerability detection, and penetration testing. Some technologies can be repurposed. For example, instead of using explainable AI to validate malware detection, it can be used to hide artifacts. And some technologies can be inverted. For example, an insider detection model can be used to help cover tracks and avoid detection. To help raise awareness, we recommend that researchers note the implications of their work, even for defensive technologies. One caveat is that the 'sword' is not symmetric depending on the wielder. For example, generative AI (deepfakes) is better for the attacker, but anomaly detection is better for the defender.

B. The Industry's Perspective

Using logic to automate attacks is not new to industry – for instance, in 2015, security researchers from FireEye [251] found that advanced Russian cyber threat groups built a malware called HAMMERTOSS that used rules based automation to blend its traffic into normal traffic by checking for regular office hours in the time zone and then operating only in that time range. However, the scale and speed that offensive AI capabilities can endow attackers can be damaging.

According to 2019 Verizon Data Breach report analysis of 140 security breaches [252], the meantime to compromising an organization and exfiltrating the data ranges is already in the order of minutes. Organizations are already finding it difficult to combat automated offensive tactics and anticipate attacks to get stealthier in the future. For instance, according to the final report released by the US National Security Commission on AI in 2021 [253], the warning is clear "The U.S. government is not prepared to defend the United States in the coming artificial intelligence (AI) era." The final report reasons that this is "Because of AI, adversaries will be able to act with micro-precision, but at macro-scale and with greater speed. They will use AI to enhance cyber attacks and digital disinformation campaigns and to target individuals in new ways."

Most organizations see offensive AI as an imminent threat – 49% of 102 cybersecurity organizations surveyed by Forrester market research in 2020 [254], anticipate offensive AI techniques to manifest in the next 12 months. As a result, more organizations are turning to ways to defend against these attacks. A 2021 survey [255] of 309 organizations' business leaders, C-Suite executives found that 96% of the organizations surveyed are already making investments to guard against AI-powered attacks as they anticipate more automation than what their defenses can handle.

C. What's on the Horizon

With AI's rapid pace of development and open accessibility, we expect to see a noticeable shift in attack strategies on organizations. First, we foresee that the number of deepfake phishing incidents will increase. This is because the technology (1) is mature, (2) is harder to mitigate than regular phishing, (3) is more effective at exploiting trust, (4) can expedite attacks, and (5) is new as phishing tactic so people are not expecting it. Second, we expect that AI will enable adversaries to target more organizations in parallel and more frequently. As a result, instead of being covert, adversaries may chose to overwhelm the defender's response teams with thousands of attempts for the chance of one success. Finally, as adversaries begin to use AI-enabled bots, defenders will be forced to automate their defences with bots as well. Keeping humans in the loop to control and determine high level strategies is a practical and ethical requirement. However, further discussion and research is necessary to form safe and agreeable policies.

D. What can be done?

Attacks Using AI. Industry and academia should focus on developing solutions for mitigating the top threats. Personnel can be shown what to expect from AI-powered social engineering and further research can be done on detecting deepfakes, but in a manner which is robust to a dynamic adversary [16]. Moreover, we recommend research into post-processing tools that can protect software from analysis after development (i.e., anti-vulnerability detection).

Attacks Against AI. The advantages and vulnerabilities of AI have profoundly questioned their widespread adoption, especially in mission-critical and cybersecurity-related tasks. In the meantime, organizations are working on automating the development and operations of ML models (MLOps), without focusing too much on ML security-related issues. To bridge this gap, we argue that extending the current MLOps paradigm to also encompass ML security (MLSecOps) may be a relevant way towards improving the security posture of such organizations. To this end, we envision the incorporation of security testing, protection and monitoring of AI/ML models into MLOps. Doing so will enable organizations to seamlessly deploy and maintain more secure and reliable AI/ML models.

VII. Conclusion

In this SoK we first explored, categorized, and identified the threats of offensive AI against organizations (sections II and III). We then detailed the threats and ranked them through a user study with experts from the domain (sections IV and V). Finally, we provided insights into our results and gave directions for future work (section VI). We hope this SoK will be meaningful and helpful to the community in addressing the imminent threat of offensive AI.

REFERENCES

[1] R. K. Knake, "A cyberattack on the u.s. power grid," tech. rep., Council on Foreign Relations, 2017.

[2] T. A. Mattei, "Privacy, confidentiality, and security of health care information: Lessons from the recent wannacry cyberattack," *World neurosurgery*, vol. 104, p. 972—974, August 2017.

[3] N. Tariq, "Impact of cyberattacks on financial institutions," *The Journal of Internet Banking and Commerce*, vol. 23, pp. 1–11, 2018.

[4] 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.

[5] H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," *applied sciences*, vol. 9, no. 20, p. 4396, 2019.

[6] N. A. Mahadi, M. A. Mohamed, A. I. Mohamad, M. Makhtar, M. F. A. Kadir, and M. Mamat, "A survey of machine learning techniques for behavioral-based biometric user authentication," *Recent Advances in Cryptography and Network Security*, pp. 43–54, 2018.

[7] "Deepreflect: Discovering malicious functionality through binary reconstruction," in *30th USENIX Security Symposium (USENIX Security 21)*, USENIX Association, Aug. 2021.

[8] D. Ucci, L. Aniello, and R. Baldoni, "Survey of machine learning techniques for malware analysis," *Computers & Security*, vol. 81, pp. 123–147, 2019.

[9] D. Cohen, Y. Mirsky, M. Kamp, T. Martin, Y. Elovici, R. Puzis, and A. Shabtai, "Dante: A framework for mining and monitoring darknet traffic," in *European Symposium on Research in Computer Security*, pp. 88–109, Springer, 2020.

[10] M. Barreno, B. Nelson, A. Joseph, and J. Tygar, "The security of machine learning," *Machine Learning*, vol. 81, pp. 121–148, 2010.

[11] L. Huang, A. D. Joseph, B. Nelson, B. Rubinstein, and J. D. Tygar, "Adversarial machine learning," in *4th ACM Workshop on Artificial Intelligence and Security (AISec 2011)*, (Chicago, IL, USA), pp. 43–57, 2011.

[12] A. D. Joseph, B. Nelson, B. I. P. Rubinstein, and J. Tygar, *Adversarial Machine Learning*. Cambridge University Press, 2018.

[13] B. Biggio and F. Roli, "Wild Patterns: Ten Years After the Rise of Adversarial Machine Learning," *Pattern Recognition*, vol. 84, pp. 317–331, 2018.

[14] A. Chakraborty, M. Alam, V. Dey, A. Chattopadhyay, and D. Mukhopadhyay, "Adversarial Attacks and Defences: A Survey," *arXiv:1810.00069 [cs, stat]*, vol. ACM Computing Survey, Sept. 2018. arXiv: 1810.00069.

[15] N. Papernot, P. McDaniel, A. Sinha, and M. P. Wellman, "Sok: Security and privacy in machine learning," in *2018 IEEE European Symposium on Security and Privacy (EuroS P)*, pp. 399–414, 2018.

[16] Y. Mirsky and W. Lee, "The creation and detection of deepfakes: A survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 1, pp. 1–41, 2021.

[17] C. Stupp, "Fraudsters used ai to mimic ceo's voice in unusual cybercrime case." https://www.wsj.com/articles/fraudsters-use-ai-to-mimic-ceos-voice-in-unusual-cybercrime-case-11567157402. (Accessed on 10/14/2020).

[18] I. Zelinka, S. Das, L. Sikora, and R. Šenkeřík, "Swarm virus-next-generation virus and antivirus paradigm?," *Swarm and Evolutionary Computation*, vol. 43, pp. 207–224, 2018.

[19] T. C. Truong, I. Zelinka, and R. Senkerik, "Neural swarm virus," in *Swarm, Evolutionary, and Memetic Computing and Fuzzy and Neural Computing*, pp. 122–134, Springer, 2019.

[20] M. Brundage, S. Avin, J. Clark, H. Toner, P. Eckersley, B. Garfinkel, A. Dafoe, P. Scharre, T. Zeitzoff, B. Filar, *et al.*, "The malicious use of artificial intelligence: Forecasting, prevention, and mitigation," *arXiv preprint arXiv:1802.07228*, 2018.

[21] M. Caldwell, J. Andrews, T. Tanay, and L. Griffin, "Ai-enabled future crime," *Crime Science*, vol. 9, no. 1, pp. 1–13, 2020.

[22] G. Lin, S. Wen, Q.-L. Han, J. Zhang, and Y. Xiang, "Software vulnerability detection using deep neural networks: A survey," *Proceedings of the IEEE*, vol. 108, no. 10, pp. 1825–1848, 2020.

[23] zerofox, "zerofox-oss/snap_r: A machine learning based social media pen-testing tool." https://github.com/zerofox-oss/SNAP_R, 2020. (Accessed on 10/21/2020).

[24] S. Calzavara, G. Tolomei, A. Casini, M. Bugliesi, and S. Orlando, "A supervised learning approach to protect client authentication on the web," *ACM Trans. Web*, vol. 9, June 2015.

[25] R. R. Yager, "Approximate reasoning as a basis for rule-based expert systems," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-14, no. 4, pp. 636–643, 1984.

[26] W. Zeng and R. L. Church, "Finding shortest paths on real road networks: The case for a*," *Int. J. Geogr. Inf. Sci.*, vol. 23, p. 531–543, Apr. 2009.

[27] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, (Cambridge, MA, USA), p. 2672–2680, MIT Press, 2014.

[28] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surv.*, vol. 53, June 2020.

[29] M. Hussain, A. Al-Haiqi, A. Zaidan, B. Zaidan, M. M. Kiah, N. B. Anuar, and M. Abdulnabi, "The rise of keyloggers on smartphones: A survey and insight into motion-based tap inference attacks," *Pervasive and Mobile Computing*, vol. 25, pp. 1–25, 2016.

[30] A. R. Javed, M. O. Beg, M. Asim, T. Baker, and A. H. Al-Bayatti, "Alphalogger: Detecting motion-based side-channel attack using smartphone keystrokes," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–14, 2020.

[31] P. Marquardt, A. Verma, H. Carter, and P. Traynor, "(sp) iphone: Decoding vibrations from nearby keyboards using mobile phone accelerometers," in *Proceedings of the 18th ACM conference on Computer and communications security*, pp. 551–562, 2011.

[32] Y. Abid, A. Imine, and M. Rusinowitch, "Sensitive attribute prediction for social networks users," in *EDBT/ICDT Workshops*, 2018.

[33] J. Jiang, X. Yu, Y. Sun, and H. Zeng, "A survey of the software vulnerability discovery using machine learning techniques," in *International Conference on Artificial Intelligence and Security*, pp. 308–317, Springer, 2019.

[34] S. A. Mokhov, J. Paquet, and M. Debbabi, "The use of nlp techniques in static code analysis to detect weaknesses and vulnerabilities," in *Advances in Artificial Intelligence* (M. Sokolova and P. van Beek, eds.), (Cham), pp. 326–332, Springer International Publishing, 2014.

[35] Y. Mirsky, T. Mahler, I. Shelef, and Y. Elovici, "Ct-gan: Malicious tampering of 3d medical imagery using deep learning," in *28th USENIX Security Symposium (USENIX Security 19)*, (Santa Clara, CA), pp. 461–478, USENIX Association, Aug. 2019.

[36] M. Schreyer, T. Sattarov, B. Reimer, and D. Borth, "Adversarial learning of deepfakes in accounting," 2019.

[37] B. Hitaj, P. Gasti, G. Ateniese, and F. Perez-Cruz, "Passgan: A deep learning approach for password guessing," in *International Conference on Applied Cryptography and Network Security*, pp. 217–237, Springer, 2019.

[38] V. Garg and L. Ahuja, "Password guessing using deep learning," in *2019 2nd International Conference on Power Energy, Environment and Intelligent Control (PEEIC)*, pp. 38–40, IEEE, 2019.

[39] C. Novo and R. Morla, "Flow-based Detection and Proxy-based Evasion of Encrypted Malware C2 Traffic," in *Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security*, AISec'20, (New York, NY, USA), pp. 83–91, Association for Computing Machinery, Nov. 2020.

[40] D. Han, Z. Wang, Y. Zhong, W. Chen, J. Yang, S. Lu, X. Shi, and X. Yin, "Practical traffic-space adversarial attacks on learning-based nidss," *arXiv preprint arXiv:2005.07519*, 2020.

[41] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *22nd ACM SIGKDD Int'l Conf. Knowl. Disc. Data Mining*, KDD '16, (New York, NY, USA), pp. 1135–1144, ACM, 2016.

[42] T. Rahman, M. Rochan, and Y. Wang, "Video-based person re-identification using refined attention networks," in *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–8, 2019.

[43] X. Zhu, X. Jing, X. You, X. Zhang, and T. Zhang, "Video-based person re-identification by simultaneously learning intra-video and inter-video distance metrics," *IEEE Transactions on Image Processing*, vol. 27, no. 11, pp. 5683–5695, 2018.

[44] Y. Zhang, J. E. Meng, and M. Pratama, "Extractive document summarization based on convolutional neural networks," in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 918–922, 2016.

[45] A. Castiglione, R. D. Prisco, A. D. Santis, U. Fiore, and F. Palmieri, "A botnet-based command and control approach relying on swarm intelligence," *Journal of Network and Computer Applications*, vol. 38, pp. 22–33, 2014.

[46] J. A. Bland, M. D. Petty, T. S. Whitaker, K. P. Maxwell, and W. A. Cantrell, "Machine learning cyberattack and defense strategies," *Computers & Security*, vol. 92, p. 101738, 2020.

[47] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," in *29th Int'l Conf. on Machine Learning* (J. Langford and J. Pineau, eds.), pp. 1807–1814, Omnipress, 2012.

[48] L. Muñoz-González, B. Biggio, A. Demontis, A. Paudice, V. Wongrassamee, E. C. Lupu, and F. Roli, "Towards Poisoning of Deep Learning Algorithms with Back-gradient Optimization," in *10th ACM Workshop on Artificial Intelligence and Security* (B. M. Thuraisingham, B. Biggio, D. M. Freeman, B. Miller, and A. Sinha, eds.), AISec '17, (New York, NY, USA), pp. 27–38, ACM, 2017.

[49] P. W. Koh and P. Liang, "Understanding Black-box Predictions via Influence Functions," in *International Conference on Machine Learning (ICML)*, 2017.

[50] T. Gu, B. Dolan-Gavitt, and S. Garg, "BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain," in *NIPS Workshop on Machine Learning and Computer Security*, vol. abs/1708.06733, 2017.

[51] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, "Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning," *ArXiv e-prints*, vol. abs/1712.05526, 2017.

[52] Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang, and X. Zhang, "Trojaning attack on neural networks," 2017.

[53] A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, and T. Goldstein, "Poison frogs! targeted clean-label poisoning attacks on neural networks," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, (Red Hook, NY, USA), p. 6106–6116, Curran Associates Inc., 2018.

[54] H. Aghakhani, D. Meng, Y.-X. Wang, C. Kruegel, and G. Vigna, "Bullseye Polytope: A Scalable Clean-Label Poisoning Attack with Improved Transferability," *arXiv preprint arXiv:2005.00191*, 2020.

[55] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli, "Evasion attacks against machine learning at test time," in *Machine Learning and Knowledge Discovery in Databases (ECML PKDD), Part III* (H. Blockeel, K. Kersting, S. Nijssen, and F. Železný, eds.), vol. 8190 of *LNCS*, pp. 387–402, Springer Berlin Heidelberg, 2013.

[56] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *International Conference on Learning Representations*, 2014.

[57] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *International Conference on Learning Representations*, 2015.

[58] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1528–1540, ACM, 2016.

[59] I. Shumailov, Y. Zhao, D. Bates, N. Papernot, R. Mullins, and R. Anderson, "Sponge examples: Energy-latency attacks on neural networks," *arXiv preprint arXiv:2006.03463*, 2020.

[60] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership Inference Attacks Against Machine Learning Models," in *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, May 2017.

[61] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pp. 111–125, 2008.

[62] S. Hidano, T. Murakami, S. Katsumata, S. Kiyomoto, and G. Hanaoka, "Model inversion attacks for prediction systems: Without knowledge of non-sensitive attributes," in *2017 15th Annual Conference on Privacy, Security and Trust (PST)*, pp. 115–11509, 2017.

[63] M. Juuti, S. Szyller, S. Marchal, and N. Asokan, "Prada: Protecting against dnn model stealing attacks," in *2019 IEEE European Symposium on Security and Privacy (EuroS P)*, pp. 512–527, 2019.

[64] H. Jia, C. A. Choquette-Choo, V. Chandrasekaran, and N. Papernot, "Entangled watermarks as a defense against model extraction," 2021.

[65] H. Zhang, H. Chen, Z. Song, D. S. Boning, I. S. Dhillon, and C. Hsieh, "The limitations of adversarial training and the blind-spot attack," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, OpenReview.net, 2019.

[66] S. Woh and J. Lee, "Game state prediction with ensemble of machine learning techniques," in *2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS)*, pp. 89–92, 2018.

[67] E. Bagdasaryan and V. Shmatikov, "Blind backdoors in deep learning models," 05 2020.

[68] Y. Yao, H. Li, H. Zheng, and B. Y. Zhao, "Latent backdoor attacks on deep neural networks," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, CCS '19, (New York, NY, USA), p. 2041–2055, Association for Computing Machinery, 2019.

[69] S. Wang, S. Nepal, C. Rudolph, M. Grobler, S. Chen, and T. Chen, "Backdoor attacks against transfer learning with pre-trained deep learning models," *IEEE Transactions on Services Computing*, pp. 1–1, 2020.

[70] J. Breier, X. Hou, D. Jap, L. Ma, S. Bhasin, and Y. Liu, "Practical fault attack on deep neural networks," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2204–2206, 2018.

[71] A. Demontis, M. Melis, M. Pintor, M. Jagielski, B. Biggio, A. Oprea, C. Nita-Rotaru, and F. Roli, "Why Do Adversarial Attacks Transfer? Explaining Transferability of Evasion and Poisoning Attacks," in *28th USENIX Security Symposium (USENIX Security 19)*, USENIX Association, 2019.

[72] M. Melis, A. Demontis, M. Pintor, A. Sotgiu, and B. Biggio, "secml: A python library for secure and explainable machine learning," *arXiv preprint arXiv:1912.10013*, 2019.

[73] M.-I. Nicolae, M. Sinn, M. N. Tran, B. Buesser, A. Rawat, M. Wistuba, V. Zantedeschi, N. Baracaldo, B. Chen, H. Ludwig, I. Molloy, and B. Edwards, "Adversarial robustness toolbox v1.2.0," *CoRR*, vol. 1807.01069, 2018.

[74] N. Papernot, F. Faghri, N. Carlini, I. Goodfellow, R. Feinman, A. Kurakin, C. Xie, Y. Sharma, T. Brown, A. Roy, A. Matyasko, V. Behzadan, K. Hambardzumyan, Z. Zhang, Y.-L. Juang, Z. Li, R. Sheatsley, A. Garg, J. Uesato, W. Gierke, Y. Dong, D. Berthelot, P. Hendricks, J. Rauber, and R. Long, "Technical report on the cleverhans v2.1.0 adversarial examples library," *arXiv preprint arXiv:1610.00768*, 2018.

[75] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *ICML*, 2020.

[76] B. Krebs, "Target hackers broke in via hvac company – krebs on security." https://krebsonsecurity.com/2014/02/target-hackers-broke-in-via-hvac-company/, 2014. (Accessed on 04/15/2021).

[77] B. I. Messaoud, K. Guennoun, M. Wahbi, and M. Sadik, "Advanced persistent threat: new analysis driven by life cycle phases and their challenges," in *2016 International Conference on Advanced Communication Systems and Information Security (ACOSIS)*, pp. 1–6, IEEE, 2016.

[78] J. Chen, C. Su, K.-H. Yeh, and M. Yung, "Special issue on advanced persistent threat," 2018.

[79] A. Alshamrani, S. Myneni, A. Chowdhary, and D. Huang, "A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1851–1877, 2019.

[80] K. Leetaru, "Deep fakes' greatest threat is surveillance video." https://www.forbes.com/sites/kalevleetaru/2019/08/26/deep-fakes-greatest-threat-is-surveillance-video/?sh=73c35a6c4550, 2019. (Accessed on 04/15/2021).

[81] M. H. Abd El-Jawad, R. Hodhod, and Y. M. K. Omar, "Sentiment analysis of social media networks using machine learning," in *2018 14th International Computer Engineering Conference (ICENCO)*, pp. 174–176, 2018.

[82] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, and M. Gruteser, "Snooping keystrokes with mm-level audio ranging on a single phone," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pp. 142–154, 2015.

[83] I. Shumailov, L. Simon, J. Yan, and R. Anderson, "Hearing your touch: A new acoustic side channel on smartphones," *arXiv preprint arXiv:1903.11137*, 2019.

[84] J. V. Monaco, "What are you searching for? a remote keylogging attack on search engine autocomplete," in *28th USENIX Security Symposium (USENIX Security 19)*, (Santa Clara, CA), pp. 959–976, USENIX Association, Aug. 2019.

[85] A. Malhotra, L. Totti, W. Meira Jr., P. Kumaraguru, and V. Almeida, "Studying user footprints in different online social networks," in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, ASONAM '12, (USA), p. 1065–1070, IEEE Computer Society, 2012.

[86] H. Hajipour, M. Malinowski, and M. Fritz, "Ireen: Iterative reverse-engineering of black-box functions via neural program synthesis," 2020.

[87] S. Datta, "Deepobfuscode: Source code obfuscation through sequence-to-sequence networks," 2020.

[88] J. Li, L. Zhou, H. Li, L. Yan, and H. Zhu, "Dynamic Traffic Feature Camouflaging via Generative Adversarial Networks," in *2019 IEEE Conference on Communications and Network Security (CNS)*, pp. 268–276, June 2019.

[89] "Our work with the dnc: Setting the record straight," Jun 2020.

[90] D. Lunghi, J. Horejsi, and C. Pernet, "Untangling the patchwork cyberespionage group," 2017.

[91] R. Leong, D. Perez, and T. Dean, "Messagetap: Who's reading your text messages?," 2019.

[92] E. Brumaghin, H. Unterbrink, and E. Tacheau, "Old dog, new tricks - analysing new rtf-based campaign distributing agent tesla, loki with pyrebox." https://blog.talosintelligence.com/2018/10/old-dog-new-tricks-analysing-new-rtf_15.html, 2018.

[93] L. Arsene, "Oil gas spearphishing campaigns drop agent tesla spyware in advance of historic opec+ deal," 2020.

[94] X. Zhang, "Analysis of new agent tesla spyware variant," 2018.

[95] R. Mueller, "Indictment - united states of america vs. viktor borisovich netyksho, et al.," 2018.

[96] G. Beni, "Swarm intelligence," *Complex Social and Behavioral Systems: Game Theory and Agent-Based Models*, pp. 791–818, 2020.

[97] H. Otsuka, Y. Watanabe, and Y. Matsumoto, "Learning from before and after recovery to detect latent misconfiguration," in *2015 IEEE 39th Annual Computer Software and Applications Conference*, vol. 3, pp. 141–148, IEEE, 2015.

[98] W. Chen, X. Qiao, J. Wei, H. Zhong, and X. Huang, "Detecting inter-component configuration errors in proactive: a relation-aware method," in *2014 14th International Conference on Quality Software*, pp. 184–189, IEEE, 2014.

[99] M. Yousefi, N. Mtetwa, Y. Zhang, and H. Tianfield, "A reinforcement learning approach for attack graph analysis," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 212–217, 2018.

[100] R. Wu, J. Gong, W. Tong, and B. Fan, "Network attack path selection and evaluation based on q-learning," *Applied Sciences*, vol. 11, no. 1, 2021.

[101] X. Ou, S. Govindavajhala, and A. W. Appel, "MulVAL: A Logic-based Network Security Analyzer," in *USENIX Security Symposium*, 2005.

[102] M. Matta, G. C. Cardarilli, L. Di Nunzio, R. Fazzolari, D. Giardino, M. re, F. Silvestri, and S. Spanò, "Q-rts: a real-time swarm intelligence based on multi-agent q-learning," *Electronics Letters*, 03 2019.

[103] Y. Leviathan and Y. Matias, "Google duplex: an ai system for accomplishing real-world tasks over the phone," 2018.

[104] S. Singh and H. K. Thakur, "Survey of various ai chatbots based on technology used," in *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, pp. 1074–1079, IEEE, 2020.

[105] Y. Rebryk and S. Beliaev, "Convoice: Real-time zero-shot voice style transfer with convolutional network," *arXiv preprint arXiv:2005.07815*, 2020.

[106] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, z. Chen, P. Nguyen, R. Pang, I. Lopez Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in Neural Information Processing Systems 31* (S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), pp. 4480–4490, Curran Associates, Inc., 2018.

[107] N. O. Leslie, R. E. Harang, L. P. Knachel, and A. Kott, "Statistical models for the number of successful cyber intrusions," *CoRR*, vol. abs/1901.04531, 2019.

[108] A. Kumar, A. Biswas, and S. Sanyal, "ecommercegan : A generative adversarial network for e-commerce," 2018.

[109] M. Manning, G. T. Wong, T. Graham, T. Ranbaduge, P. Christen, K. Taylor, R. Wortley, T. Makkai, and P. Skorich, "Towards a 'smart'cost–benefit tool: using machine learning to predict the costs of criminal justice policy interventions," *Crime Science*, vol. 7, no. 1, p. 12, 2018.

[110] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital twin: Enabling technologies, challenges and open research," *IEEE Access*, vol. 8, pp. 108952–108971, 2020.

[111] R. Bitton, T. Gluck, O. Stan, M. Inokuchi, Y. Ohta, Y. Yamada, T. Yagyu, Y. Elovici, and A. Shabtai, "Deriving a cost-effective digital twin of an ics to facilitate security evaluation," in *European Symposium on Research in Computer Security*, pp. 533–554, Springer, 2018.

[112] G. Pasandi, S. Nazarian, and M. Pedram, "Approximate logic synthesis: A reinforcement learning-based technology mapping approach," in *20th International Symposium on Quality Electronic Design (ISQED)*, pp. 26–32, IEEE, 2019.

[113] H. Anderson, "Evading machine learning malware detection," 2017.

[114] Z. Fang, J. Wang, B. Li, S. Wu, Y. Zhou, and H. Huang, "Evading anti-malware engines with deep reinforcement learning," *IEEE Access*, vol. 7, pp. 48867–48879, 2019.

[115] T. Perianin, S. Carré, V. Dyseryn, A. Facon, and S. Guilley, "End-to-end automated cache-timing attack driven by machine learning," *Journal of Cryptographic Engineering*, pp. 1–12, 2020.

[116] P. Mozur, "Looking through the eyes of china's surveillance state," 2018. accessed: June 2018.

[117] Y. Wang, T. Bao, C. Ding, and M. Zhu, "Face recognition in real-world surveillance videos with deep learning method," in *Image, Vision and Computing (ICIVC), 2017 2nd International Conference on*, pp. 239–243, IEEE, 2017.

[118] C. Ding, K. Huang, V. M. Patel, and B. C. Lovell, "Special issue on video surveillance-oriented biometrics," *Pattern Recognition Letters*, vol. 107, pp. 1–2, 2018.

[119] P. Bontrager, A. Roy, J. Togelius, N. Memon, and A. Ross, "Deepmasterprints: Generating masterprints for dictionary attacks via latent variable evolution," in *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pp. 1–9, IEEE, 2018.

[120] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee, *et al.*, "The asvspoof 2019 database," *arXiv preprint arXiv:1911.01601*, 2019.

[121] D. Wang, A. Neupane, Z. Qian, N. B. Abu-Ghazaleh, S. V. Krishnamurthy, E. J. Colbert, and P. Yu, "Unveiling your keystrokes: A cache-based side-channel attack on graphics libraries.," in *NDSS*, 2019.

[122] X. Liu, Z. Zhou, W. Diao, Z. Li, and K. Zhang, "When good becomes evil: Keystroke inference with smartwatch," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 1273–1285, 2015.

[123] A. Maiti, M. Jadliwala, J. He, and I. Bilogrevic, "Side-channel inference attacks on mobile keypads using smartwatches," *IEEE Transactions on Mobile Computing*, vol. 17, no. 9, pp. 2180–2194, 2018.

[124] A. Compagno, M. Conti, D. Lain, and G. Tsudik, "Don't skype & type! acoustic eavesdropping in voice-over-ip," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 703–715, 2017.

[125] J. Yu, L. Lu, Y. Chen, Y. Zhu, and L. Kong, "An indirect eavesdropping attack of keystrokes on touch screen through acoustic sensing," *IEEE Transactions on Mobile Computing*, 2019.

[126] L. Lu, J. Yu, Y. Chen, Y. Zhu, X. Xu, G. Xue, and M. Li, "Keylisterber: Inferring keystrokes on qwerty keyboard of touch screen through acoustic signals," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 775–783, IEEE, 2019.

[127] Y. Chen, T. Li, R. Zhang, Y. Zhang, and T. Hedgpeth, "Eyetell: Video-assisted touchscreen keystroke inference from eye movements," in *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 144–160, IEEE, 2018.

[128] Y. Wang, W. Cai, T. Gu, W. Shao, I. Khalil, and X. Xu, "Gazerevealer: Inferring password using smartphone front camera," in *Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pp. 254–263, 2018.

[129] Y. Wang, W. Cai, T. Gu, and W. Shao, "Your eyes reveal your secrets: an eye movement based password inference on smartphone," *IEEE transactions on mobile computing*, 2019.

[130] J. Sun, X. Jin, Y. Chen, J. Zhang, Y. Zhang, and R. Zhang, "Visible: Video-assisted keystroke inference from tablet backside motion.," in *NDSS*, 2016.

[131] K. S. Balagani, M. Conti, P. Gasti, M. Georgiev, T. Gurtler, D. Lain, C. Miller, K. Molas, N. Samarin, E. Saraci, *et al.*, "Silk-tv: Secret information leakage from keystroke timing videos," in *European Symposium on Research in Computer Security*, pp. 263–280, Springer, 2018.

[132] J. Lim, T. Price, F. Monrose, and J.-M. Frahm, "Revisiting the threat space for vision-based keystroke inference attacks," *arXiv preprint arXiv:2009.05796*, 2020.

[133] S. Nam, S. Jeon, H. Kim, and J. Moon, "Recurrent gans password cracker for iot password security enhancement," *Sensors*, vol. 20, no. 11, p. 3106, 2020.

[134] J. Seymour and P. Tully, "Generative models for spear phishing posts on social media," *arXiv preprint arXiv:1802.05196*, 2018.

[135] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Annual international cryptology conference*, pp. 388–397, Springer, 1999.

[136] L. Lerman, G. Bontempi, and O. Markowitch, "Power analysis attack: an approach based on machine learning," *International Journal of Applied Cryptography*, vol. 3, no. 2, pp. 97–115, 2014.

[137] K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analysis: Concrete results," in *International workshop on cryptographic hardware and embedded systems*, pp. 251–261, Springer, 2001.

[138] D. Brumley and D. Boneh, "Remote timing attacks are practical," *Computer Networks*, vol. 48, no. 5, pp. 701–716, 2005.

[139] L. Lerman, G. Bontempi, S. B. Taieb, and O. Markowitch, "A time series approach for profiling attack," in *International Conference on Security, Privacy, and Applied Cryptography Engineering*, pp. 75–94, Springer, 2013.

[140] L. Weissbart, S. Picek, and L. Batina, "One trace is all it takes: Machine learning-based side-channel attack on eddsa," in *Security, Privacy, and Applied Cryptography Engineering* (S. Bhasin, A. Mendelson, and M. Nandi, eds.), (Cham), pp. 86–105, Springer International Publishing, 2019.

[141] S. Picek, A. Heuser, A. Jovic, S. Bhasin, and F. Regazzoni, "The curse of class imbalance and conflicting metrics with machine learning for side-channel evaluations," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2019, no. 1, pp. 1–29, 2019.

[142] E. Cagli, C. Dumas, and E. Prouff, "Convolutional neural networks with data augmentation against jitter-based countermeasures," in *International Conference on Cryptographic Hardware and Embedded Systems*, pp. 45–68, Springer, 2017.

[143] A. Heuser, S. Picek, S. Guilley, and N. Mentens, "Side-channel analysis of lightweight ciphers: Does lightweight equal easy?," in *International Workshop on Radio Frequency Identification: Security and Privacy Issues*, pp. 91–104, Springer, 2016.

[144] S. Picek, I. P. Samiotis, J. Kim, A. Heuser, S. Bhasin, and A. Legay, "On the performance of convolutional neural networks for side-channel analysis," in *International Conference on Security, Privacy, and Applied Cryptography Engineering*, pp. 157–176, Springer, 2018.

[145] H. Maghrebi, T. Portigliatti, and E. Prouff, "Breaking cryptographic implementations using deep learning techniques," in *International Conference on Security, Privacy, and Applied Cryptography Engineering*, pp. 3–26, Springer, 2016.

[146] G. Perin, Chmielewski, L. Batina, and S. Picek, "Keep it unsupervised: Horizontal attacks meet deep learning," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2021, pp. 343–372, Dec. 2020.

[147] D. X. Song, D. A. Wagner, and X. Tian, "Timing analysis of keystrokes and timing attacks on ssh.," in *USENIX Security Symposium*, vol. 2001, 2001.

[148] E. C. R. Shin, D. Song, and R. Moazzezi, "Recognizing functions in binaries with neural networks," in *24th {USENIX} Security Symposium ({USENIX} Security 15)*, pp. 611–626, 2015.

[149] X. Xu, C. Liu, Q. Feng, H. Yin, L. Song, and D. Song, "Neural network-based graph embedding for cross-platform binary code similarity detection," *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, Oct 2017.

[150] T. Bao, J. Burket, M. Woo, R. Turner, and D. Brumley, "{BYTEWEIGHT}: Learning to recognize functions in binary code," in *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, pp. 845–860, 2014.

[151] B. Liu, W. Huo, C. Zhang, W. Li, F. Li, A. Piao, and W. Zou, "αdiff: cross-version binary code similarity detection with dnn," in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pp. 667–678, 2018.

[152] S. H. Ding, B. C. Fung, and P. Charland, "Asm2vec: Boosting static representation robustness for binary clone search against code obfuscation and compiler optimization," in *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 472–489, IEEE, 2019.

[153] Y. Duan, X. Li, J. Wang, and H. Yin, "Deepbindiff: Learning program-wide code representations for binary diffing," in *Proceedings of the 27th Annual Network and Distributed System Security Symposium (NDSS'20)*, 2020.

[154] F. Ye, S. Zhou, A. Venkat, R. Marucs, N. Tatbul, J. J. Tithi, P. Petersen, T. Mattson, T. Kraska, P. Dubey, *et al.*, "Misim: An end-to-end neural code similarity system," *arXiv preprint arXiv:2006.05265*, 2020.

[155] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, "Graph transformer networks," *arXiv preprint arXiv:1911.06455*, 2019.

[156] T. Huybrechts, Y. Vanommeslaeghe, D. Blontrock, G. Van Barel, and P. Hellinckx, "Automatic reverse engineering of can bus data using machine learning techniques," in *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, pp. 751–761, Springer, 2017.

[157] H. Li, B. Shuai, J. Wang, and C. Tang, "Protocol reverse engineering using lda and association analysis," in *2015 11th International Conference on Computational Intelligence and Security (CIS)*, pp. 312–316, IEEE, 2015.

[158] G. Bossert, F. Guihéry, and G. Hiet, "Towards automated protocol reverse engineering using semantic information," in *Proceedings of the 9th ACM symposium on Information, computer and communications security*, pp. 51–62, 2014.

[159] Y. Wang, Z. Zhang, D. D. Yao, B. Qu, and L. Guo, "Inferring protocol state machine from network traces: a probabilistic approach," in *International Conference on Applied Cryptography and Network Security*, pp. 1–18, Springer, 2011.

[160] Q. Feng, R. Zhou, C. Xu, Y. Cheng, B. Testa, and H. Yin, "Scalable graph-based bug search for firmware images," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 480–491, 2016.

[161] Z. Li, D. Zou, S. Xu, X. Ou, H. Jin, S. Wang, Z. Deng, and Y. Zhong, "Vuldeepecker: A deep learning-based system for vulnerability detection," *arXiv preprint arXiv:1801.01681*, 2018.

[162] Z. Li, D. Zou, J. Tang, Z. Zhang, M. Sun, and H. Jin, "A comparative study of deep learning-based vulnerability detection system," *IEEE Access*, vol. 7, pp. 103184–103197, 2019.

[163] S. Chakraborty, R. Krishna, Y. Ding, and B. Ray, "Deep learning based vulnerability detection: Are we there yet?," *arXiv preprint arXiv:2009.07235*, 2020.

[164] D. She, K. Pei, D. Epstein, J. Yang, B. Ray, and S. Jana, "Neuzz: Efficient fuzzing with neural program smoothing," in *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 803–817, IEEE, 2019.

[165] D. She, R. Krishna, L. Yan, S. Jana, and B. Ray, "Mtfuzz: Fuzzing with a multi-task neural network," *arXiv preprint arXiv:2005.12392*, 2020.

[166] Y. Wang, P. Jia, L. Liu, C. Huang, and Z. Liu, "A systematic review of fuzzing based on machine learning techniques," *PloS one*, vol. 15, no. 8, p. e0237749, 2020.

[167] Y. Li, S. Ji, C. Lyu, Y. Chen, J. Chen, Q. Gu, C. Wu, and R. Beyah, "V-fuzz: Vulnerability prediction-assisted evolutionary fuzzing for binary programs," *IEEE Transactions on Cybernetics*, 2020.

[168] L. Cheng, Y. Zhang, Y. Zhang, C. Wu, Z. Li, Y. Fu, and H. Li, "Optimizing seed inputs in fuzzing with machine learning," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, pp. 244–245, IEEE, 2019.

[169] V. Atlidakis, R. Geambasu, P. Godefroid, M. Polishchuk, and B. Ray, "Pythia: Grammar-based fuzzing of rest apis with coverage-guided feedback and learning-based mutations," *arXiv preprint arXiv:2005.11498*, 2020.

[170] M. Janota, "Towards generalization in qbf solving via machine learning.," in *AAAI*, pp. 6607–6614, 2018.

[171] H. Samulowitz and R. Memisevic, "Learning to solve qbf," in *AAAI*, vol. 7, pp. 255–260, 2007.

[172] J. H. Liang, C. Oh, M. Mathew, C. Thomas, C. Li, and V. Ganesh, "Machine learning-based restart policy for cdcl sat solvers," in *International Conference on Theory and Applications of Satisfiability Testing*, pp. 94–110, Springer, 2018.

[173] V. Kurin, S. Godil, S. Whiteson, and B. Catanzaro, "Improving sat solver heuristics with graph networks and reinforcement learning," *arXiv preprint arXiv:1909.11830*, 2019.

[174] C. Martorella, "laramies/metagoofil: Metadata harvester." https://github.com/laramies/metagoofil, 2020. (Accessed on 10/20/2020).

[175] Y. Ghazi, Z. Anwar, R. Mumtaz, S. Saleem, and A. Tahir, "A supervised machine learning based approach for automatically extracting high-level threat intelligence from unstructured sources," in *2018 International Conference on Frontiers of Information Technology (FIT)*, pp. 129–134, IEEE, 2018.

[176] J. Guo, Y. Fan, L. Pang, L. Yang, Q. Ai, H. Zamani, C. Wu, W. B. Croft, and X. Cheng, "A deep look into neural ranking models for information retrieval," *Information Processing & Management*, p. 102067, 2019.

[177] L. Akoglu, H. Tong, and D. Koutra, "Graph based anomaly detection and description: a survey," *Data mining and knowledge discovery*, vol. 29, no. 3, pp. 626–688, 2015.

[178] J. Schwartz and H. Kurniawati, "Autonomous penetration testing using reinforcement learning," *arXiv preprint arXiv:1905.05965*, 2019.

[179] Oxylabs, "Innovative proxy service to gather data at scale." https://oxylabs.io/, 2021. (Accessed on 04/14/2021).

[180] R. Dabre, C. Chu, and A. Kunchukuttan, "A survey of multilingual neural machine translation," *ACM Computing Surveys (CSUR)*, vol. 53, no. 5, pp. 1–38, 2020.

[181] Z. Nasar, S. W. Jaffry, and M. K. Malik, "Textual keyword extraction and summarization: State-of-the-art," *Information Processing & Management*, vol. 56, no. 6, p. 102088, 2019.

[182] J. R. G. Evangelista, R. J. Sassi, M. Romero, and D. Napolitano, "Systematic literature review to investigate the application of open source intelligence (osint) with artificial intelligence," *Journal of Applied Security Research*, pp. 1–25, 2020.

[183] "Telegram contest." https://github.com/IlyaGusev/tgcontest. (Accessed on 10/14/2020).

[184] I. Ilin, "Building a news aggregator from scratch: news filtering, classification, grouping in threads and ranking." https://towardsdatascience.com/building-a-news-aggregator-from-scratch-news-filtering-classification-grouping-in-threads-and-7b0bbf619b68. (Accessed on 10/14/2020).

[185] B. Wang and N. Z. Gong, "Stealing hyperparameters in machine learning," in *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 36–52, IEEE, 2018.

[186] L. Batina, S. Bhasin, D. Jap, and S. Picek, "CSI NN: Reverse engineering of neural network architectures through electromagnetic side channel," in *28th USENIX Security Symposium (USENIX Security 19)*, (Santa Clara, CA), pp. 515–532, USENIX Association, Aug. 2019.

[187] J. Breier, D. Jap, X. Hou, S. Bhasin, and Y. Liu, "Sniff: Reverse engineering of neural networks with fault attacks," *arXiv preprint arXiv:2002.11021*, 2020.

[188] B. Zhou, M. Elbadry, R. Gao, and F. Ye, "Batmapper: Acoustic sensing based indoor floor plan construction using smartphones," in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, pp. 42–55, 2017.

[189] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128837–128868, 2019.

[190] Y. Ren, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Almost unsupervised text to speech and automatic speech recognition," *arXiv preprint arXiv:1905.06791*, 2019.

[191] A. White, A. Matthews, K. Snow, and F. Monrose, "Phonotactic reconstruction of encrypted voip conversations: Hookt on fon-iks," pp. 3 – 18, 06 2011.

[192] A. Al-Hababi and S. C. Tokgoz, "Man-in-the-middle attacks to detect and identify services in encrypted network flows using machine learning," in *2020 3rd International Conference on Advanced Communication Technologies and Networking (CommNet)*, pp. 1–5, IEEE, 2020.

[193] Y. Nirkin, Y. Keller, and T. Hassner, "Fsgan: Subject agnostic face swapping and reenactment," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7184–7193, 2019.

[194] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "First order motion model for image animation," in *Conference on Neural Information Processing Systems (NeurIPS)*, December 2019.

[195] M. Workman, "Wisecrackers: A theory-grounded investigation of phishing and pretext social engineering threats to information security," *Journal of the American Society for Information Science and Technology*, vol. 59, no. 4, pp. 662–674, 2008.

[196] J. Salminen, S.-g. Jung, and B. J. Jansen, "The future of data-driven personas: A marriage of online analytics numbers and human attributes.," in *ICEIS (1)*, pp. 608–615, 2019.

[197] J. Salminen, R. G. Rao, S.-g. Jung, S. A. Chowdhury, and B. J. Jansen, "Enriching social media personas with personality traits: A deep learning approach using the big five classes," in *International Conference on Human-Computer Interaction*, pp. 101–120, Springer, 2020.

[198] D. Spiliotopoulos, D. Margaris, and C. Vassilakis, "Data-assisted persona construction using social media data," *Big Data and Cognitive Computing*, vol. 4, no. 3, p. 21, 2020.

[199] S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng, and B. Y. Zhao, "Fawkes: Protecting privacy against unauthorized deep learning models," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pp. 1589–1604, 2020.

[200] Q. Sun, A. Tewari, W. Xu, M. Fritz, C. Theobalt, and B. Schiele, "A hybrid model for identity obfuscation by face replacement," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 553–569, 2018.

[201] Y. Li, X. Yang, B. Wu, and S. Lyu, "Hiding faces in plain sight: Disrupting ai face synthesis with adversarial perturbations," *arXiv preprint arXiv:1906.09288*, 2019.

[202] shaoanlu, "shaoanlu/faceswap-gan: A denoising autoencoder + adversarial losses and attention mechanisms for face swapping.." https://github.com/shaoanlu/faceswap-GAN, 2020. (Accessed on 10/19/2020).

[203] Q. Wang, W. Zhao, J. Yang, J. Wu, W. Hu, and Q. Xing, "Deeptrust: A deep user model of homophily effect for trust prediction," in *2019 IEEE International Conference on Data Mining (ICDM)*, pp. 618–627, IEEE, 2019.

[204] R. Kong and X. Tong, "Dynamic weighted heuristic trust path search algorithm," *IEEE Access*, vol. 8, pp. 157382–157390, 2020.

[205] S. Roller, E. Dinan, N. Goyal, D. Ju, M. Williamson, Y. Liu, J. Xu, M. Ott, K. Shuster, E. M. Smith, Y.-L. Boureau, and J. Weston, "Recipes for building an open-domain chatbot," *arXiv:2004.13637 [cs]*, Apr. 2020. arXiv: 2004.13637.

[206] J. Seymour and P. Tully, "Weaponizing data science for social engineering: Automated e2e spear phishing on twitter," *Black Hat USA*, vol. 37, pp. 1–39, 2016.

[207] A. Das and R. Verma, "Automated email generation for targeted attacks using natural language," 2019.

[208] Z. Fu, X. Tan, N. Peng, D. Zhao, and R. Yan, "Style transfer in text: Exploration and evaluation," *arXiv preprint arXiv:1711.06861*, 2017.

[209] Z. Yang, Z. Hu, C. Dyer, E. P. Xing, and T. Berg-Kirkpatrick, "Unsupervised text style transfer using language models as discriminators," in *Advances in Neural Information Processing Systems*, pp. 7287–7298, 2018.

[210] A. Panagiotou, B. Ghita, S. Shiaeles, and K. Bendiab, "Facewallgraph: Using machine learning for profiling user behaviour from facebook wall," in *Internet of Things, Smart Spaces, and Next Generation Networks and Systems* (O. Galinina, S. Andreev, S. Balandin, and Y. Koucheryavy, eds.), (Cham), pp. 125–134, Springer International Publishing, 2019.

[211] C. Dhaoui, C. M. Webster, and L. P. Tan, "Social media sentiment analysis: lexicon versus machine learning," *Journal of Consumer Marketing*, 2017.

[212] M. Ghiassi and S. Lee, "A domain transferable lexicon set for twitter sentiment analysis using a supervised machine learning approach," *Expert Systems with Applications*, vol. 106, pp. 197–216, 2018.

[213] M. Rathi, A. Malik, D. Varshney, R. Sharma, and S. Mendiratta, "Sentiment analysis of tweets using machine learning approach," in *2018 Eleventh International Conference on Contemporary Computing (IC3)*, pp. 1–3, 2018.

[214] "Black Hat USA 2018."

[215] H. Pellet, S. Shiaeles, and S. Stavrou, "Localising social network users and profiling their movement," *Computers & Security*, vol. 81, pp. 49–57, 2019.

[216] W. Zhang, R. Lau, S. Liao, and R.-W. Kwok, "A probabilistic generative model for latent business networks mining," *International Conference on Information Systems, ICIS 2012*, vol. 2, pp. 1102–1118, 01 2012.

[217] Z. Ma, O. Sheng, and G. Pant, "Discovering company revenue relations from news: A network approach," *Decision Support Systems*, vol. 47, pp. 408–414, 11 2009.

[218] A. Kumar and N. Rathore, "Improving attribute inference attack using link prediction in online social networks," in *Recent Advances in Mathematics, Statistics and Computer Science*, pp. 494–503, World Scientific, 2016.

[219] M. Zhang and Y. Chen, "Link prediction based on graph neural networks," in *Advances in Neural Information Processing Systems*, pp. 5165–5175, 2018.

[220] Q. Cao, Y. Qiao, and Z. Lyu, "Machine learning to detect anomalies in web log analysis," in *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, pp. 519–523, 2017.

[221] B. Debnath, M. Solaimani, M. A. G. Gulzar, N. Arora, C. Lumezanu, J. Xu, B. Zong, H. Zhang, G. Jiang, and L. Khan, "Loglens: A real-time log analysis system," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, pp. 1052–1062, 2018.

[222] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *International Conference on Machine Learning*, pp. 201–210, 2016.

[223] J. Breier, X. Hou, D. Jap, L. Ma, S. Bhasin, and Y. Liu, "Deeplaser: Practical fault attack on deep neural networks," *arXiv preprint arXiv:1806.05859*, 2018.

[224] S. Li, S. Ma, M. Xue, and B. Z. H. Zhao, "Deep learning backdoors," *arXiv preprint arXiv:2007.08273*, 2020.

[225] K. Hasegawa, M. Yanagisawa, and N. Togawa, "Trojan-net classification for gate-level hardware design utilizing boundary net structures," *IEICE TRANSACTIONS on Information and Systems*, vol. 103, no. 7, pp. 1618–1622, 2020.

[226] B. Kolosnjaji, A. Demontis, B. Biggio, D. Maiorca, G. Giacinto, C. Eckert, and F. Roli, "Adversarial Malware Binaries: Evading Deep Learning for Malware Detection in Executables," in *26th European Signal Processing Conf.*, EUSIPCO, (Rome), pp. 533–537, IEEE, 2018.

[227] A. Demontis, M. Melis, M. Pintor, M. Jagielski, B. Biggio, A. Oprea, C. Nita-Rotaru, and F. Roli, "Why Do Adversarial Attacks Transfer? Explaining Transferability of Evasion and Poisoning Attacks," in *28th USENIX Security Symposium (USENIX Security 19)*, USENIX Association, 2019.

[228] D. Maiorca, A. Demontis, B. Biggio, F. Roli, and G. Giacinto, "Adversarial Detection of Flash Malware: Limitations and Open Issues," *Computers & Security*, vol. 96, p. 101901, 2020.

[229] K. K. Ispoglou and M. Payer, "malwash: Washing malware to evade dynamic analysis," in *10th USENIX Workshop on Offensive Technologies (WOOT 16)*, (Austin, TX), USENIX Association, Aug. 2016.

[230] O. Suciu, S. E. Coull, and J. Johns, "Exploring adversarial examples in malware detection," in *2019 IEEE Security and Privacy Workshops (SPW)*, pp. 8–14, 2019.

[231] L. Demetrio, B. Biggio, G. Lagorio, F. Roli, and A. Armando, "Functionality-preserving Black-box Optimization of Adversarial Windows Malware," *arXiv:2003.13526 [cs]*, Sept. 2020. arXiv: 2003.13526.

[232] 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, May 2020.

[233] H. S. Anderson, A. Kharkar, B. Filar, D. Evans, and P. Roth, "Learning to evade static pe machine learning malware models via reinforcement learning," 2018.

[234] F. Zhiyang, J. Wang, B. Li, S. Wu, Y. Zhou, and H. Huang, "Evading anti-malware engines with deep reinforcement learning," *IEEE Access*, vol. PP, pp. 1–1, 03 2019.

[235] A. C. Bahnsen, I. Torroledo, L. D. Camacho, and S. Villegas, "Deepphish: Simulating malicious ai," in *2018 APWG Symposium on Electronic Crime Research (eCrime)*, pp. 1–8, 2018.

[236] J. Peck, C. Nie, R. Sivaguru, C. Grumer, F. Olumofin, B. Yu, A. Nascimento, and M. De Cock, "Charbot: A simple and effective method for evading dga classifiers," *IEEE Access*, vol. 7, pp. 91759–91771, 2019.

[237] L. Sidi, A. Nadler, and A. Shabtai, "Maskdga: An evasion attack against dga classifiers and adversarial defenses," *IEEE Access*, vol. 8, pp. 161580–161592, 2020.

[238] Y. Sharon, D. Berend, Y. Liu, A. Shabtai, and Y. Elovici, "Tantra: Timing-based adversarial network traffic reshaping attack," *arXiv preprint arXiv:2103.06297*, 2021.

[239] A. G. Sutro, "Machine-Learning Based Evaluation of Access Control Lists to Identify Anomalies," Jan. 2020.

[240] N. Dalvi, P. Domingos, Mausam, S. Sanghai, and D. Verma, "Adversarial classification," in *Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, (Seattle), pp. 99–108, 2004.

[241] D. Lowd and C. Meek, "Adversarial learning," in *Proc. 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, (Chicago, IL, USA), pp. 641–647, ACM Press, 2005.

[242] D. Lowd and C. Meek, "Good word attacks on statistical spam filters," in *Second Conference on Email and Anti-Spam (CEAS)*, (Mountain View, CA, USA), 2005.

[243] J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi, "Black-box generation of adversarial text sequences to evade deep learning classifiers," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 50–56, 2018.

[244] Y. Li, Y. Wang, Y. Wang, L. Ke, and Y.-a. Tan, "A feature-vector generative adversarial network for evading pdf malware classifiers," *Information Sciences*, vol. 523, pp. 38–48, 2020.

[245] B. Biggio, I. Corona, G. Fumera, G. Giacinto, and F. Roli, "Bagging classifiers for fighting poisoning attacks in adversarial classification tasks," in *10th International Workshop on Multiple Classifier Systems (MCS)* (C. Sansone, J. Kittler, and F. Roli, eds.), vol. 6713 of *Lecture Notes in Computer Science*, pp. 350–359, Springer-Verlag, June 2011.

[246] M. Rigaki and S. Garcia, "Bringing a gan to a knife-fight: Adapting malware communication to avoid detection," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 70–75, 2018.

[247] M. I. Patel, S. Suthar, and J. Thakar, "Survey on image compression using machine learning and deep learning," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 1103–1105, IEEE, 2019.

[248] M. Abadi and D. G. Andersen, "Learning to Protect Communications with Adversarial Neural Cryptography," *arXiv*, 2016.

[249] M. Guri and Y. Elovici, "Bridgeware: The air-gap malware," *Communications of the ACM*, vol. 61, no. 4, pp. 74–82, 2018.

[250] S. Jiang, D. Ye, J. Huang, Y. Shang, and Z. Zheng, "Smartsteganogaphy: Light-weight generative audio steganography model for smart embedding application," *Journal of Network and Computer Applications*, vol. 165, p. 102689, 2020.

[251] F. E. T. Intelligence, "Hammertoss: stealthy tactics define a russian cyber threat group," *Milpitas, CA: FireEye, Inc*, 2015.

[252] "2019 data breach investigations report," *Verizon, Inc*, 2019.

[253] "Final report - national security commission on artificial intelligence," *National Security Commission on Artificial Intelligence*, 2021.

[254] "The emergence of offensive ai," *Forrester*, 2020.

[255] "Preparing for ai-enabled cyberattacks," *MIT Technology Review Insights*, 2021.