Y. Mirsky, A. Demontis, J. Kotak, R. Shankar, D. Gelei, L. Yang, X. Zhang, M. Pintor, W. Lee, Y. Elovici, and B. Biggio. The threat of offensive AI to organizations. Computers & Security, 124:103006, 2023.

https://doi.org/10.1016/j.cose.2022.103006

The Threat of Offensive AI to Organizations

Yisroel Mirsky[1]

Ben-Gurion University of the Negev, Israel

Ambra Demontis

University of Cagliari, Italy

Jaidip Kotak

Ben-Gurion University of the Negev, Israel

Ram Shankar

Microsoft, United States of America

Deng Gelei

Nanyang Technological University, Singapore

Liu Yang

Nanyang Technological University, Singapore

Xiangyu Zhang

Purdue University, United States of America

Maura Pintor

University of Cagliari & Pluribus One, Italy

Wenke Lee

Georgia Institute of Technology, United States of America

Yuval Elovici

Ben-Gurion University of the Negev, Israel

Battista Biggio

University of Cagliari & Pluribus One, Italy

Email addresses: yisroel@post.bgu.ac.il (Yisroel Mirsky), ambrademontis@gmail.com (Ambra Demontis), jaidip@post.bgu.ac.il (Jaidip Kotak), ramk@microsoft.com (Ram Shankar), gelei.deng@ntu.edu.sg (Deng Gelei), yangliu@ntu.edu.sg (Liu Yang), xyzhang@cs.purdue.edu (Xiangyu Zhang), maura.pintor@unica.it (Maura Pintor), wenke@cc.gatech.edu (Wenke Lee), elovici@bgu.ac.il (Yuval Elovici), battista.biggio@diee.unica.it (Battista Biggio)

[1]Corresponding Author

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 study, 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 32 offensive AI capabilities which adversaries can use to enhance their attacks. Finally, through a panel survey spanning industry, government and academia, we rank the AI threats and provide insights on the adversaries.

Keywords: Offensive AI, APT, cyber security, organization security, adversarial machine learning, deepfake, AI-capable adversary

1. Introduction

For decades, organizations, including government agencies, hospitals, and financial institutions, have been the target of cyber attacks [1, 2, 3]. These cyber attacks have been carried out by experienced hackers using manual methods. 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[2] in the context of improving the state of cyber defense [4, 5, 6] and threat analysis [7, 8, 9]. However, AI is a double edged sword, and attackers can utilize it to improve their malicious campaigns.

Therefore, we define Offensive AI as

> *"The use or abuse of AI to accomplish a malicious task"*

Offensive Use of AI. 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 their employer's face

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

and voice [10, 11]. It is also possible to improve the stealth capabilities of attacks by enabling them to proceed without human supervision and aid (making it automatic). For example, if malware could perform a progressive infection of hosts in a network (a.k.a., lateral movement) on its own, then this would reduce command and control (C&C) communication [12, 13]. 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 3).

Offensive Abuse of AI. Adversarial machine learning is the study of security vulnerabilities in AI. It has been shown that an adversary can craft training samples to alter the functionalities of a model e.g., insert a backdoor [14], obtain a desired classification manipulating the test samples (e.g., evade detection) [15] and even infer confidential information about a model [16] or the data on which it was trained [17]. Since organizations use AI to automate the management, maintenance, operation and defence of their systems and services, an adversary can accomplish their malicious goals by using machine learning *offensively* on these systems (adversarial machine learning).

We note that some attacks are achievable without using or abusing AI. However, attackers can substantially reduce the effort required to perform an attack if they use AI to make it automatic or semi-automatic. By reducing their effort in creating effective strategies, attackers can maximize their return by scaling the attacks in their strength and quantity. Moreover, by acting simultaneously in several phases of the attack chain, the attacker can achieve synergistic effects on the speed and power of the attacks, becoming even more dangerous. On the other hand, some attacks have been enabled by AI, such as the cloning of an individual's voice in a sophisticated social engineering attack [18].

1.1. Study Overview

In this work, we provide a study of knowledge 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 [19] the authors warned the community that AI can be used for unethical and criminal purposes with examples taken from various domains. In [20] a workshop was held that attempted to identify the potential top threats of AI in criminology. However, both these works relate to the threat of AI on society overall and are not specific to organizations and their networks. Moreover, despite their efforts and preliminary results, these previous analyses provide only examples of how AI can be used to attack and a possible ranking of their risk, while our study gives a structured view of offensive AI through the standard methodologies used to identify potential attack tactics against organizations, deriving strategic insights relevant to defend from these threats.

To accomplish these goals, we performed a literature review to identify the capabilities of an AI-capable adversary. We then performed a panel survey to identify which of these capabilities represent the most relevant threats in practice. There were 35 survey participants: 16 from academia and 19 from industry. The participants from industry were from a wide profile of organizations such as MITRE, IBM, Microsoft, Google, Airbus, Bosch, Fujitsu, Hitachi, and Huawei.

From our literature review, we identified 32 offensive AI capabilities against organizations. Our panel survey revealed that the most significant threats are the capabilities that improve social engineering attacks (e.g., the use of deepfakes to clone the voice of employees). We also found that industry members are most concerned about attacks that enable attackers to steal intellectual property and detect vulnerabilities in their software. Finally, we have also found that modern offensive AI mainly impacts the initial steps of the cyber kill chain (reconnaissance, resource development, and initial access). This is because AI technologies are not mature enough to create agents able to carry on attacks that proceed without human supervision and aid. A complete list of our findings can be found in section 5.1.

1.2. Contributions

In this study, 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 2.3).
- An enumeration and description of the 32 offensive AI capabilities that threaten organizations, based on literature review and current events (section 3). These capabilities can be categorised as (1) automation, (2) campaign resilience, (3) credential theft, (4) exploit development, (5) information gathering, (6) social engineering, and (7) stealth.
- A threat ranking and insights on how offensive AI impacts organizations, based on a panel survey with members from academia, industry, and government (section 4).
- A forecast of the AI threat horizon and the resulting shifts in attack strategies (section 5).

1.3. Article Structure

This article is structured as follows:

- In section 2, we provide the reader with a primer on topics which are important for understanding the literature review. The section introduces concepts about AI, offensive AI, and how offensive AI impacts an organization's security.
- In section 3, we offer our literature review of offensive AI in the context of an organization's security.
- In section 4, we present the results from a panel survey to help identify the least and most significant threats of offensive AI to organizations.

- In section 5, we summarize our findings and provide our observations on the matter.

2. Background

In this section, we provide the reader with technical aspects related to offensive AI and introduce offensive AI concepts related to organizations' security. Later in section 3, we review the latest research on the topic.

2.1. AI and Machine Learning

AI is a larger domain that mainly deals with creating algorithms that can automate complex tasks. Early AI models were rule-based systems designed using an expert's knowledge [21], followed by search algorithms for selecting optimal decisions (e.g., finding paths or playing games [22]). Today, the most popular type of AI is machine learning (ML), which is a data-driven approach to AI where programs automatically improve their performance on a task-given experience. 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 industrial automation, forecasting, and planning due to its ability to reason upon and generate complex data. Due to the popularity of ML, our literature review inevitably follows this trend. Despite considering all methods and techniques related to using AI in general, we found the vast majority of the offensive AI techniques we found use ML to perform AI-based attacks. Therefore, the majority of the works reviewed in this study involve some form of ML.

In general, a machine learning model can be trained on data with 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 paradigm 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 [23] 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.

In the context of offensive AI, the location in which an attacker performs training or execution will depend on the attacker's objective and strategy. For example, for reconnaissance tasks, training and execution will likely take place offsite from the organization. However, for attacks, the training and execution may take place onsite, offsite, or both. Another possibility is where the adversary uses few-shot learning [24] by training on general data offsite and then fine tuning on the target data onsite. Additional examples can be found in Table 1.

Table 1: Examples of where a model can be trained and executed in an attack on an organization. Onsite refers to being within the premisis or network of the 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

In all cases, the adversary will first design and evaluate their model offsite prior to its usage in the organization to ensure its success and 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.

2.2. Offensive AI

As noted in section 1, there are two forms of offensive AI (OAI): Attacks using AI and attacks abusing AI. For example, an adversary can (1) use 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 OAI is commonly referred to as adversarial machine learning.

We will now elaborate on these two forms of offensive AI.

2.2.1. Attacks Using AI

Although there are a wide variety of AI tasks which can be used in attacks, we now list the most common ones. Note that these tasks are not mutually exclusive, in fact some build on each other and produce a synergistic effect on their impact on the attack chain.

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 [25] 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.

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 [26] and the use of heuristic attack graphs to plan optimal attacks on networks [27].

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 [28, 29], intelligent password guessing [30, 31], and traffic shaping to avoid detection [32, 33]. 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 [10].

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 [34, 35, 36], the selection of the weakest link in the chain to attack [37], and the localization of software vulnerabilities for exploitation [38, 39, 40].

Retrieval This is the task of finding content that matches or that is semantically similar 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 [41, 42], to find a disgruntled employee (as a potential insider) using semantic analysis on social media posts, and to summarize lengthy documents [43] during open source intelligence (OSINT) gathering in the reconnaissance phase.

2.2.2. Attacks Abusing AI

An attacker can use its AI knowledge to exploit ML model vulnerabilities violating its confidentiality, integrity, or availability [15]. The vast majority of these attacks is studied in Adversarial Machine Learning, a branch of research that investigates on how to obtain specific malfunctions on ML models to create malicious attacks. These 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 [44, 45, 46] are when the attacker decreases the model's performance until it is unusable. A backdoor poisoning attack [14, 47] or trojaning attack [48], 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 [49, 50]

Modify the Test Data. In this case, the attacker modifies test samples to have them misclassified [51, 52, 53]. 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 [54]. 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. [55].

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 [56], deanonymization [57], and model inversion [58] attacks. For learning about the model's parameters there are model stealing/extraction [59, 60], and blind-spot detection [61], state prediction [62].

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, compromising a loss (training) function to insert a backdoor [63] or slowing down the created model [64].

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 tampers its parameters to insert a latent behavior. These attacks can be performed at the software [65, 66, 66] or hardware [67] 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.
- **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 analyze the model's responses in a query-based manner to get feedback on certain inputs.
- **Gray-Box (Limited-Knowledge) Attacks:** The attacker has partial knowledge of the target system (e.g., the learning algorithm, architecture, etc.,).

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. [51, 68].

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 [69, 70, 71, 72].

2.3. 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 enterprises. 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 3, we will detail the adversary's techniques based on our literature review.

2.3.1. Attacker Motivation

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 offensively on an organization. From our literature review (detailed later in section 3), we observed three reasons why an adversary may be motivated to use offensive AI against an organization: coverage, speed, and success.

Coverage. By using AI, an adversary can scale up its operations by automating complex tasks to decrease human labor and increase the chances of success. For example, AI can be used to automatically craft [10, 11] and launch (employing [73, 74, 75]) spear phishing attacks, distill [43] data collected from OSINT, and reach more assets within a network [76, 77] 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 [78, 79], intelligently select the next best target during lateral movement [80], spy on users to obtain information (e.g., perform speech to text on eavesdropped audio) [81], or find zero-days in software [38, 39, 40]. 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 or camouflaging network traffic (such as C2 traffic) [32, 33] and by exploiting weaknesses in the defender's AI models such as an ML-based intrusion detection system (IDS) [82], (2) identify opportunities such as good targets for social engineering attacks [37] and novel vulnerabilities [38, 39, 40], (3) enable better attack vectors such as using deepfakes in spear phishing attacks [11], (4) plan optimal attack strategies [27, 80], and (5) strengthen persistence in the network through automated bot coordination [26] and malware obfuscation [83].

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.

2.3.2. 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 [84]. These agents not only pose a threat to the organization, but also to its employees, customers, and the general public as well (e.g., attacks on critical infrastructure).

In an attack, there may be a number of attack steps that the threat agent must accomplish. These steps depend on the adversary's goal and strategy. For example, in an advanced persistent threat (APT) [85, 86, 87], 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 a 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 that involve multiple steps can be thwarted if the defender identifies or blocks the attack early on. The more progress that 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 to shorten and obscure the kill chain 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 server (C2). This generates presence in the defenders network which can be detected over time.

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 of AI technology increases.

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

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

2.3.3. *New Threats*

AI-capable adversaries have new abilites over conventional cyber adversaries. These abilities give attackers the means to novel acts of sabotage, espionage and theft of intellectual property (IP):

Sabotage. An adversary can use its knowledge to cause damage to an organization in ways that weren't possible before. This is because AI-based adversaries can use (1) adversarial machine learning, (2) generative AI, and (3) deep learning for software analysis.

With adversarial machine learning, an attacker can target the organization's ML products and solutions. For example, they can poison datasets to harm an ML model's performance or plant a backdoor in a model for later exploitation. More examples include, the ability to evade detection in surveillance [54] and affect forecasts models (e.g., finance [88], energy [89], etc.) With generative AI, an attacker can add or modify evidence in a realistic manner. Examples include the modification of surveillance footage to include or omit evidence [90], the tampering of medical scans to harm patients [28], and the manipulation of financial records to perform fraud [29]. Finally, with recent advances in deep learning, attackers can efficiently and effectively locate vulnerabilities in both source code [91, 92] and compiled code [93, 39, 94]. This enables attackers to locate new vulnerabilities for exploitation with minimal effort.

Espionage. With AI, adversaries can spy on organizations in new ways using side-channel analysis and swarm intelligence. Side channels are signals emitted from a device that can be used to infer confidential information [95] In the past, side-channel attacks were mainly performed in labs using expensive electronics and analytical processes. With AI, adversaries can now perform side-channel attacks on-site and extract information from channels that are temporal, complex, and multi-modal. For example, a compromised smartphone can be used to automatically collect and organize conversations as text using speech-to-text (STT) algorithms, and sentiment analysis [96]. Attackers can also steal credentials through acoustic and motion side channels [97, 98]. AI can also be used to extract latent information from encrypted web traffic [99], and track users through the organization's social media [100]. Finally, by using swarm intelligence-based malware [12], attackers can minimize the number of communications that they have to make to maintain and control and progress the attack. Doing so makes it harder for the organization to detect the attacker's presence (i.e., less anomalous outbound traffic) and to remove the malware after blocking the attacker's communication lines.

IP Theft. An AI-capable adversary can extract IP from organizations in new ways. For example, ML models can be stolen from purchased software products, or from cloud services querying the models with crafted inputs [59, 60]. Similar attacks can be performed to steal the model's training data [58, 101]. Obtaining this IP can help an adversary evade or control these models whether they're deployed in the organization or another provider. Another example is AI-based reverse engineering, where compiled software is lifted into higher levels of code so that the algorithms and logic can be understood and stolen [102].

2.3.4. OAI Attack Capabilities

Using the literature review (details later in section 3), we grouped the papers according to the offensive capability they provide. Doing so revealed 32 offensive AI capabilities (OAC) which directly improve the adversary's ability to achieve attack steps (e.g., impersonation, user tracking, etc). We then grouped the OACs into categories according to their offensive activity (e.g., social engineering). Finally, we used real use cases reported in the news and by MITRE to validate the OACs and verify that none were missed.

The seven OAC categories were: (1) automation, (2) campaign resilience, (3) credential theft, (4) exploit development, (5) information gathering, (6) social engineering, and (7) stealth. These categories capture the main intent of the adversary reflecting the motivators introduced in section 2.3.1. Therefore, these categories are non-exclusive (e.g., automating intelligence gathering involves capabilities from both 'automation' and 'information gathering').

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 the attacker's ability to achieve the given attack step. These edges were obtained by (1) observing real cases reported by MITRE and academic articles and (2) mapping the cases and articles to their respective OACs and their impact on the cyber kill chain. From the figure, we can see that offensive AI impacts every aspect of the attack model. Later in section 3 we will discuss each of these 32 OACs in greater detail.

These capabilities are materialized in one of two ways:

AI-based tools are programs that perform a specific task in the adversary's arsenal. For example, a tool for intelligently predicting passwords [30, 31], obfuscating malware code [83], traffic shaping for evasion [103, 32, 33], puppeting a persona [10], and so on. These tools are typically in the form of a machine learning model.

AI-driven bots are autonomous bots that 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 [26] and machine learning to operate.

3. Literature Review

In section 2.3.4 we presented the 32 offensive AI capabilities. We will now present our literature review of the OACs in order of their 7 categories: automation, campaign resilience, credential theft, exploit development, information gathering, social engineering, and stealth.

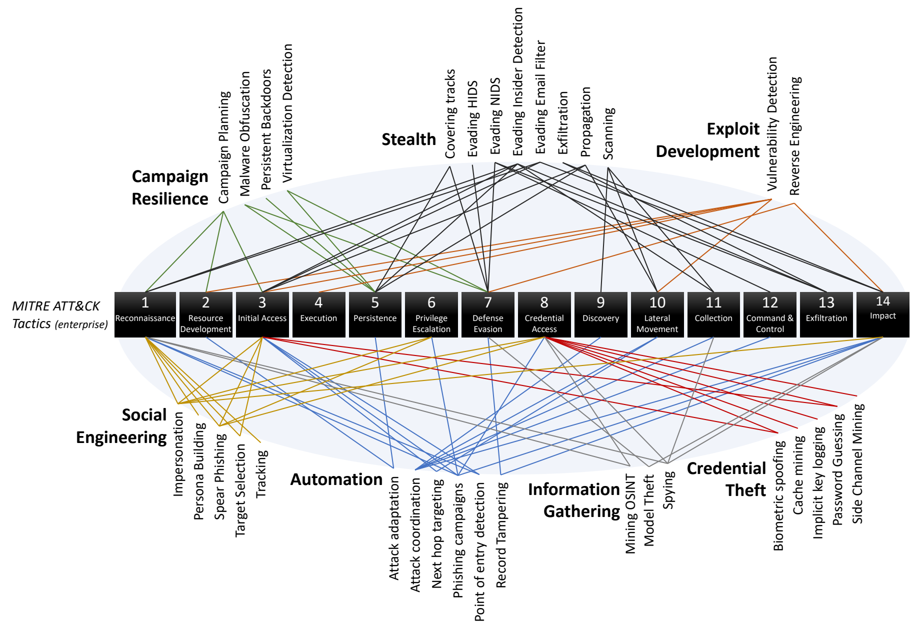

Figure 1: The 32 offensive AI capabilities (OAC) identified in our literature review, mapped to the MITRE enterprise ATT&CK model. An edge indicates that the OAC directly helps the attacker achieve the indicated attack step.

Methodology. To perform our literature review, we used the MITRE ATT&CK[5] matrix as a guide. This matrix lists the common tactics (or attack steps) that an adversary performs when attacking an organization, from planning and reconnaissance leading to the final goal of exploitation. We divided up the work among five different academic workgroups from different international institutions. Each workgroup was assigned a set of tactics from the MITRE ATT&CK matrix, based on their expertise. During the survey, the workgroups were asked to evaluate how AI has been and can be used by an attacker to improve an attacker's tactics and techniques. Finally, the workgroups cross inspected each other's content to ensure correctness and completeness.

To identify potential articles and sources to include in our literature review, we selected articles written in the English language and published in peer-reviewed international conference proceedings and journals on the topics of cybersecurity and AI from 1999. As for AI topics, we also included publicly-accessible preprint publications as well since they are well known to be the source of the latest advances from key researchers. When searching for attacks which involve AI, we used variations of both 'AI' and 'machine learning' as keywords. The selection process resulted in 225 scientific papers, from which we

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

performed our literature review.

3.1. Automation

The ability to automate complex tasks 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 that are less dependent on C2 signals. Attack automation takes form of either (1) tools which can perform complex tasks using AI (e.g., clone voices, suggest a target) or (2) software (bots) which can operate autonomously to complete an entire attack step with our human intervention (e.g., a bot/malware which propagates on its own by making decisions based on the environment or cooperatively in communication with other bots).

3.1.1. 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 [104] before attempting an exploit to increase the chances of success and avoid detection. In Black Hat'18, IBM researchers showed how malware can trigger itself using DL by identifying a target's machine by analyzing the victim's face, voice, and other attributes. With models such as decision trees, malware can locate and identify assets via complex rules like [105, 106]. Instead of transferring screenshots [107, 108, 109, 110] DL can be used onsite to extract critical information.

3.1.2. Attack Coordination

Cooperative bots can use AI to find the best times and targets to attack. For example, swarm intelligence [111] is the study of autonomous coordination among bots in a decentralized manner. Researchers have proposed that botnets can use swarm intelligence as well. In [12] the authors discuss a hypothetical swarm malware and in [13] 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).

3.1.3. 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 [112, 113]. 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 [27] where attackers and defenders are modeled as competing players. Another approach is to use DL [114, 115] to explore "attack graphs" [76] 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 [77].

3.1.4. 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 [73, 74, 75]. 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 [116], and then (3) calls the victim in the colleague's voice to exploit their trust.

3.1.5. Point of Entry Detection

The adversary can use AI to identify and select the best attack vector for an initial infection. For example, in [117] 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.

3.1.6. Record Tampering

An adversary may use AI to tamper with records as part of their end goal. For example, ML can be used to impact business decisions with synthetic data [118], to obstruct justice by tampering evidence [90], to perform fraud [29] or to modify medical or satellite imagery [28]. As shown in [28], DL-tampered records can fool human observers and can be accomplished autonomously onsite.

3.2. 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 the reuse of tools and exploits for future and parallel campaigns. AI can be used to improve campaign resilience through planning, persistence, obfuscation, and detection of virtualization to avoid dynamic analysis.

3.2.1. 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 [119], 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 [84]. Moreover, ML can be used to plan a digital twin [120, 121] of the victim's network (based on information from reconnaissance) to be created offsite for tuning AI models and developing malware.

3.2.2. Persistent Access

An adversary can have bots establish multiple back doors per host and coordinate reinfection efforts among a swarm [12]. Doing so achieves a foothold in 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 that uses ML to identify when to deploy the malware and avoid detection [122, 123]. Moreover, a USB-sized neural compute stick[6] can be planted by an insider to enable covert and autonomous onsite DL operations.

3.2.3. Malware Obfuscation

ML models such as GANs can be used to obscure a malware's intent from an analyst. Doing so can enable the 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 [83] uses recurrent neural networks (RNN) to generate ciphered code. Alternatively, backdoors can be planted in open source projects and hidden using similar manners [124].

3.2.4. 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 [125]) and other system properties.

3.3. 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.

3.3.1. Biometric spoofing

Biometric security is used for access to terminals (such as smartphones) and for performing automated surveillance [126, 127, 128]. 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) [129]. Face recognition systems can be fooled or evaded with the use of adversarial samples. For example, in [54] where the authors generated colorful glasses that alter the perceived identity. Moreover, 'sponge' samples [55] 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 [130], and by cloning the target's voice with deep learning [130].

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

3.3.2. Cache mining

Information on credentials can be found in a system's cache and log dumps, but a large amount of data makes finding it a difficult task. However, the authors of [78] showed how ML could be used to identify credentials in cache dumps from graphic libraries. Another example is the work of [79] where an ML system was used to identify cookies containing session information.

3.3.3. 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 come 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. Malware can use the smartphone's motion sensors to decipher the touch strokes on the phone [34, 35] and keystrokes on nearby keyboards [36]. Wearable devices can be exploited in a similar way as well [131, 132].

Audio. Researchers have shown that, when pressed, each key gives its own unique sound which can be used to infer what is being typed [97, 133]. The timing between keystrokes 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 [98, 134, 135].

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 [136, 137, 138], device motion [139], and hand motion [140, 141].

3.3.4. 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 [30]. Researchers have improved on this approach by using RNNs in the generation process [142]. However, the authors of [31] found that models like [30] do not work well on Russian passwords. Instead, adversaries may pass the GAN personal information on the user to improve the performance [143].

3.3.5. 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 [144, 145], electromagnetic emanations [146], processing time [147], cache hits/misses[125]. In general, ML can be used to mine nearly any kind of side channel [148, 149, 150, 151, 152, 153, 154, 155]. For example, credentials can be extracted from the timing of network traffic [156].

3.4. 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 that they can exploit.

3.4.1. 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 [157, 158, 159, 160, 161, 162, 163] 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 [164], similar to semantic analysis in language processing. Protocols and state machines can also be reversed using ML, for example, CAN bus data in vehicles [165], network protocols [166], and commands [167, 168].

3.4.2. 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 [40, 169, 91, 170, 171]. 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 [172, 173, 94, 174, 38, 175, 176]. Many works have also shown how AI can mitigate the issue of symbolic execution's massive state space [177, 178, 179, 180, 39].

3.5. 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 distill actionable intel for a campaign.

3.5.1. 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 [181], ML can be used to minimize interactions by prioritizing sites and data elements [182, 183].

Gathering. Network structure and elements can be identified using cluster analysis or graph-based anomaly detection [184]. Credentials and asset information can be found using methods like reinforcement learning on other organizations [185]. Finally, personnel structure can be extracted from social media using NLP-based web scrappers like Oxylabs[186].

Extraction. Techniques like NLP can be used to translate foreign documents [187], identify relevant documents [188, 189], extract relevant information from online sources [190, 191], and locate valid identifiers[100].

3.5.2. 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 [56, 57, 58], or (3) use it to perform a white-box attack against an organization. As described in section 2.2.2, if the model can be queried (e.g., model as a service -MAAS), then its parameters [59, 60] and hyperparameters [192] can be copied by observing the model's responses. This can also be done through side-channel [193] or hardware-level analysis [194].

3.5.3. 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 [195], (2) using object recognition [196] to obtain physical penetration info (control terminals, locks, guards, etc.), and (3) automatically mine relevant information from overheard conversations [197, 188]. ML can also be used to analyze encrypted traffic. For example it can extract transcripts from encrypted voice calls [81], identify applications [198], and reveal internet searches [99].

3.6. Social Engineering

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

3.6.1. Impersonation (Identity Theft)

An adversary may want to impersonate someone for a scam, blackmail attempt, 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 the existing media content of a victim [10]. Recently, the technology has advanced to the state where reenactment can be performed in real-time [199], and training only requires a few images [200] or seconds of audio [116] from the victim. For high-quality deepfakes, large amounts of audio/video data are 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) [201]. Moreover, the audio/video data may be an end goal inside the organization (e.g., customer data).

3.6.2. Persona Building

Adversaries build fake personas on online social networks (OSN) to connect with their targets [202]. To evade fake profile detectors, a profile can be cloned and slightly altered using AI [203, 204, 205] 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 [206, 207, 208, 209]. To build connections, a link prediction model can be used to maximize the acceptance rate [210, 211] and a DL chatbot can be used to maintain the conversations [212].

3.6.3. Spear Phishing

Call-based spear phishing attacks can be enhanced using real-time deepfakes of someone the victim trusts. For example, this occurred in 2019 when a CEO was scammed out of $240k [11]. For text-based phishing, tweets [213] and emails [214, 143, 215] can be generated to attract a specific victim, or style transfer techniques can be used to mimic a colleague [216, 217].

3.6.4. Target Selection

An adversary can use AI to identify victims in the organization who are the most susceptible to social engineering attacks [37]. It is also possible to build a model based on the target's social attributes (conversations, attended events, etc.) [218, 219]. Moreover, sentiment analysis can be used to find disgruntled employees to be recruited as insiders [220, 96, 221, 222, 223].

3.6.5. 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 [100] and through facial recognition [224]. ML models can also be used on OSN content to track a member's location [225]. Finally, ML can also be used to discover hidden business relationships [226, 227] from the news and OSNs as well [228, 229].

3.7. Stealth

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

3.7.1. Covering tracks

To hide traces of the adversary's presence, anomaly detection can be performed on the logs to remove abnormal entries [230, 231]. CryptoNets [232] 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 [67, 233] and software [48, 234] levels. DL hardware trojans can use adversarial machine learning to avoid being detected [235].

3.7.2. *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 [236, 237, 238]. For example, adversaries can evade an ML-based HIDS that performs dynamic analysis by splitting the malware's code into small components executed by different processes [239]. They can also evade ML-based detectors that perform static analysis by adding bytes to the executable [240] or code that does not affect the malware behavior [241, 242, 243, 244, 123]. Modifying the malware without breaking its malicious functionality is not easy. Attackers may use AI explanation tools like LIME [25] 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 [7].

3.7.3. *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 [245]. Bots trying to contact their C2 server can generate URLs that appear legitimate to humans [246], or that can evade malicious-URL detectors[82]. To evade traffic-based NIDSs, adversaries can shape their traffic [32, 33] or change their timing to hide it[247].

3.7.4. *Evading Insider Detectors*

To avoid insider detection mechanisms, adversaries can mask their operations using ML. For example, given one user's credentials, they can use the information on the user's role and the organization's structure to ensure that the operation performed looks legitimate [248].

3.7.5. *Evading Email Filter*

Many email services use machine learning to detect malicious emails. However, adversaries can use adversarial machine learning to evade detection [249, 250, 251, 252]. Similarly, malicious documents attached to emails, containing malware, can evade detection as well (e.g., [253]). 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 [254].

3.7.6. *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 [103] or by encoding the traffic within a permissible channel like Facebook chat [255]. To hide the transfer better, an adversary could use DL to compress [256] and even encrypt [257] the data being

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

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) [258] then DL techniques can be used to hide data within side channels such as noise in audio [259].

3.7.7. Propagation & Scanning

For stealthy lateral movement, an adversary can configure their Petri nets or attack graphs (see section 3.1.3) to avoid assets and subnets with certain IDSs and favor 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 [103].

4. Panel Survey & Threat Ranking

In our literature review (section 3), we identified the potential offensive AI capabilities (OAC) that an adversary could use to attack an organization. However, some OACs may be impractical, whereas others may pose much larger threats. Therefore, we performed a panel survey to rank these threats and understand their impact on the cyber kill chain.

4.1. Survey Setup

We surveyed 35 experts in both subjects of AI and cybersecurity. To be included in the panel survey, a participant must (1) be actively working in academia, industry or government and (1) have at least 2 years experience in *both* cybersecurity and AI.

From the industry and government sectors, we had 19 participants. Amoun then were a CISO of a large institution, a CTO and founder of AI-based security companies, an AI ethics researcher from a cybersecurity company, two research managers involved in cyber security AI projects, and seven researchers working in cybersecurity or AI-based cybersecurity. From academia, we had 16 participants: 8 professors and 8 research scientists (Ph.D. and above) with experience in both AI and cyber security. 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 (I2R), Google, Robust Intelligence, Pluribus One, Ermes Cyber Security, Mandiant, WiData, Purdue University, Georgia Institute of Technology, Munich Research Center, University of Cagliari, University of Venice, King's College London, Technische Universität Braunschweig, 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 that 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. Prior to filling out the questionnaire, all participants were given context of how offensive AI threatens organisations. Prior to rating the aspects of an OAC, participants were given one or more example instances of the OAC for clarification. The questions and the example instances can be found in the appendix. The survey was facilitated using a Google form and it took the participants approximately 30-60 minutes each to complete the form. The responses from the survey were used to produce threat rankings and to gain insights into the threat of offensive AI to organizations.

Only 35 individuals participated in the survey because AI-cybersecurity experts are very busy and hard to reach. However, given the diversity of the participants, we believe that these results still provide meaningful insights into the opinions and concerns that members of academia and industry have on offensive AI.

4.2. 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[8] in the range of 1-7 (low to high):

Profit (P): The amount of benefit that a threat agent gains by using AI compared to using non-AI methods. For example, attack success, flexibility, coverage, attack 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. This measure also includes the monetary cost to the attacker.

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 that 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 attack success. Finally, to model threat, we must consider the amount of harm

[8]The aspects are based on those proposed by [20].

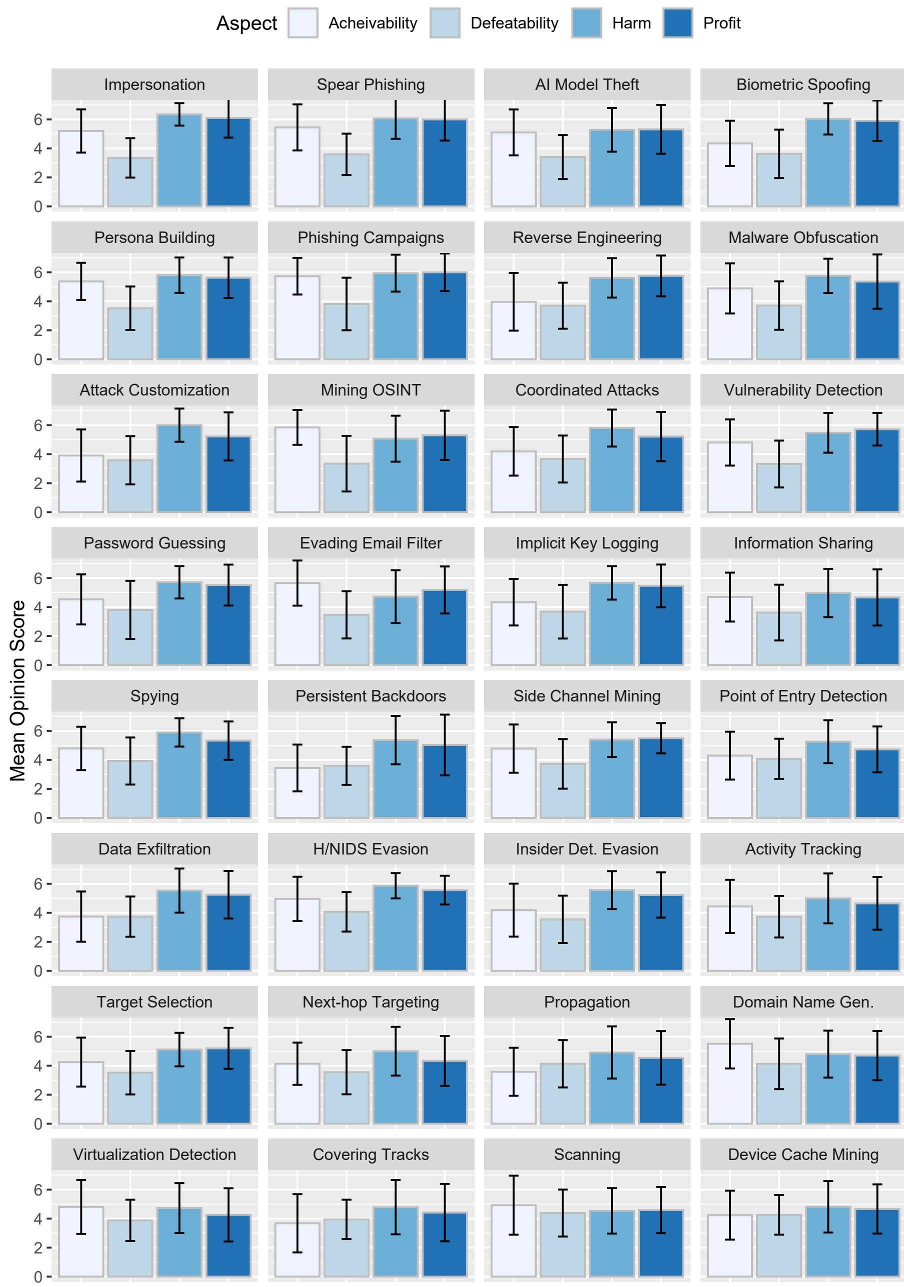

Figure 2: Survey results: the averaged and normalized opinion scores for each offensive AI capability (OAC) when used against an organization. The OACs are ordered according to their threat score, left to right, starting from the first row.

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 frequently occur so that the damage will be higher in the long run.

4.2.1. 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 19 of the OACs (60%) 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 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 is perceived to 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.

4.2.2. 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, it seems that 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

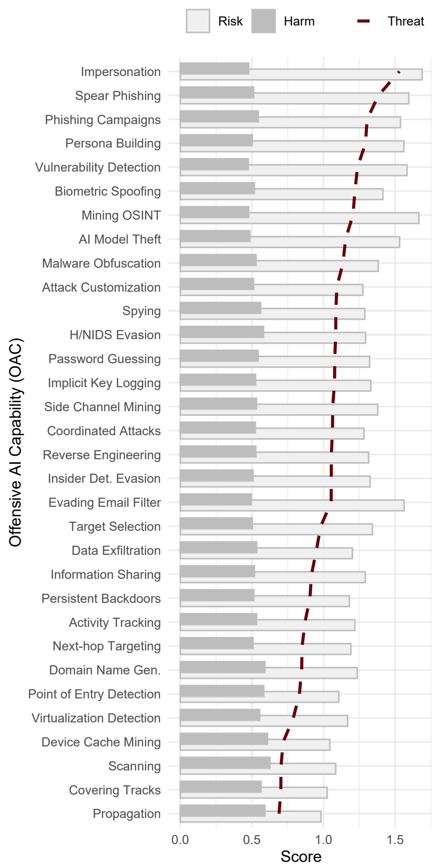

Figure 3: Survey results: the offensive AI capabilities ranked according to their threat scores.

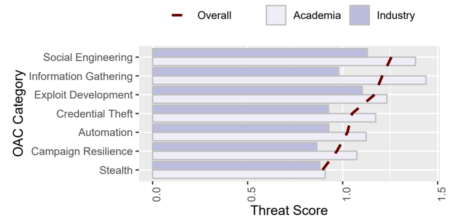

Figure 4: Survey results: the offensive AI capability categories ranked according to their average threat scores. The scores from industry and academia participants are also presented separately.

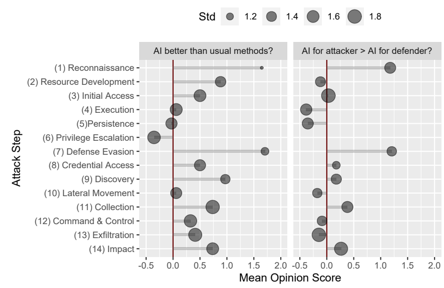

Figure 5: Survey results: Mean opinion scores on whether (1) it is more beneficial for the adversary to use AI over conventional methods, and (2) AI benefits attackers more than AI benefits defenders. The scores range from -3 to +3.

may be because these concepts have presented (proven) themselves less in the wild than the others.

Regardless, both industry and academia seem to agree on the top three most threatening OAC categories: (1) social engineering, (2) information gathering and (3) exploit development. 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, it would appear that 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 [240, 32], 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 our participants from industry and academia regarding the most threatening OACs. The top 10 most threatening OACs for organizations (out of 32) were ranked as follows:

Industry's Perspective

1. Impersonation
2. Spear Phishing
3. Phishing Campaigns
4. Persona Building
5. Vulnerability Detection
6. Reverse Engineering
7. H/NIDS Evasion
8. Mining OSINT
9. Password Guessing
10. Attack Customization

Academia's Perspective

1. Impersonation
2. Biometric Spoofing
3. Target Selection
4. Spear Phishing
5. Mining OSINT
6. Vulnerability Detection
7. Spying
8. Persona Building
9. Phishing Campaigns
10. AI Model Theft

Both industry and academia view impersonation as the greatest threat to organizations. This is understandable given recent events where deepfakes were successfully used for impersonation and fraud [260, 261, 262, 263]. We note that our participants from academia view biometric spoofing as the second largest threat, where our participants from industry don't even 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., [54, 129]). In contrast to the academics, our industry participants view 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, target selection, and the theft of AI models 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, defence evasion and social engineering, likely because these are threats which are out of their control.

4.3. Impact on the Cyber Kill Chain

For each of the 14 MITRE ATT&CK steps, we asked the participants whether they agree or disagree[9] 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. 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 prevent 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 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, which 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.

5. Findings & Discussion

In this section, we (1) present our main findings from the literature review and panel survey and (2) share our insights on our findings and discuss the road ahead.

5.1. Main Findings

From the Literature Review.

- We first observed that there are three primary motivations for an adversary to use AI: coverage, speed, and success (See Section 2.3.1).
- Offensive AI introduces new threats to organizations. A few examples include the poisoning of machine learning models [44, 14], theft of credentials through side-channel analysis [156], and the targeting of proprietary training datasets [58, 59].

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

- Adversaries can employ 32 offensive AI capabilities against organizations. These are categorized into seven groups: (1) attack 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 [38], pen-testing [213], and credential leakage detection [79] can be weaponized by adversaries for malicious purposes.

From the Panel Survey.

- The top three most threatening categories of offensive AI capabilities against organizations are (1) social engineering, (2) information gathering and (3) exploit development.

- 19 of the 32 offensive AI capabilities pose significant threats to organizations.

- 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 the highest threat.

- Aside from social engineering aspects, industry and academia are not aligned on the top threats of offensive AI against organizations. Industry members are most concerned with AI being used for reverse engineering, with a focus on the loss of intellectual property and vulnerability detection. Academics, on the other hand, are most concerned about AI being used to perform biometric spoofing (e.g., evading fingerprint and facial recognition) and attack automation.

- Although the evasion of intrusion detection systems (e.g., with adversarial machine learning) is classified as a significant threat, its only ranks number 12 on the list. This may be due to the challenge of the adversary creating effective black box attacks in an unknown IT environment.

- AI impacts the start of the cyber kill chain the most (i.e., reconnaissance, resource development, and initial access). This is because the adversary has more information available and can use this information to refine and evaluate the attacks offsite before proceeding.

- Because AI can be used 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.

5.2. 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. Humans have highly evolved and efficient perception and decision-making abilities. These rely on mental models formed throughout our lives. These mental models (like AI models) can be exploited by presenting information in ways that deceive them [264, 265]. With deepfakes, social engineering attacks have become even more frequent [10]. The same holds for software analysis where ML has been shown to be effective at analyzing software (complex structural data) whether it is source code or a compiled binary [163, 102, 92]. As mentioned earlier, we believe the reason academia is the most concerned with biometrics is that it almost exclusively uses ML, and academia is well aware of ML's flaws. Industry members may view these attacks as less threatening because physical infiltration is not a top security threat to organizations [266]. This might explain why 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 in 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 new tasks 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' networks, 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 might 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 could also be used in an offensive manner. Some technologies have a dual purpose. For example, ML research into disassembly, vulnerability detection, and penetration testing can be used for both malicious and defensive activities. Some technologies can be repurposed. For example, instead of using explainable AI to validate malware detection, it can be used to hide artifacts [267]. And some technologies can be inverted. For example, an insider detection model [248] 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 usefulness of the 'sword' is not symmetric depending on the wielder. For example, generative AI (deepfakes) might be more useful for the attacker because it allows them to generate fake samples (e.g. video) that imitate the benign ones allowing the attacker to accomplish its goal while remaining undetected. Whereas anomaly detection might be more beneficial for the defender.

Limitations of this study. Our study analyzes AI techniques that can be used by attackers against organizations through the MITRE ATT&CK Enterprise matrix. It is also important, however, to note that MITRE also offers other matrices that can be used for different use cases, namely one for Mobile[10] and one for Industrial Control Systems (ICS).[11] Although the Enterprise and Mobile tactics are almost the same, there are a few unique tactics for ICS that are not contemplated in our study, and that can be extended with the additional non-overlapping threats identified by this scenario.

5.3. The Industry's Perspective

Using logic to automate attacks is not new to industry – for instance, in 2015, security researchers from FireEye [268] 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 [269], the mean time to compromise an organization and exfiltrating the data ranges is already in the order of minutes. Organizations are already finding automated offensive tactics difficult to combat 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 [270], 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 [271], 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. In a 2021 survey [272] 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.

Presently, there are at least three nations which are actively thinking about securing ML systems: The USA through the NSCAI and NIST AI Risk Management, Frameworks[12] the UK via their recent release of Principles of securing ML systems,[13] and the EU via the EU AI act coupled with the recently proposed Cyber Resilience Act.[14] For the most part, these countries emphasise similar aspects: securing the ML pipeline and drawing attention to various attacks on AI systems. It is to be noted that all these frameworks are nascent and are

[10] https://attack.mitre.org/techniques/mobile/
[11] https://attack.mitre.org/techniques/ics/
[12] https://www.nist.gov/itl/ai-risk-management-framework
[13] https://www.ncsc.gov.uk/collection/machine-learning
[14] https://digital-strategy.ec.europa.eu/en/library/cyber-resilience-act

still under discussion. Moreover, their approach is different too. For instance, the NIST framework is voluntary but the proposed EU framework would be mandated for critical ML systems. It is a long road for these standards to come to fruition. Based on followups with our industry members, we believe that organisations may be curious at best about these frameworks but not actively adopting any at this time.

5.4. 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. In our opinion, 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 a phishing tactic so cyber defenders 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 choose 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 defenses 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 are necessary to form safe and agreeable policies.

5.5. 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 that is robust to a dynamic adversary [10]. 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 toward 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.

6. Conclusion

In this study we first explored, categorized, and identified the threats of offensive AI against organizations (sections 2 and 2.3). We then detailed the threats and ranked them through a panel survey with experts from the domain (sections 3 and 4). Finally, we provided insights into our results and gave directions for future work (section 5). We hope this study will be meaningful and helpful to the community in addressing the imminent threat of offensive AI.

7. Acknowledgments

The authors would like to thank Laurynas Adomaitis, Sin G. Teo, Manojkumar Parmar, Charles Hart, Matilda Rhode, Dr. Daniele Sgandurra, Dr. Pin-Yu Chen, Evan Downing, Didier Contis, Marco Uras, Konrad Rieck, Dr. Armin Wasicek, Dr. Stefano Traverso, and Dr. Fabio Pierazzi for taking the time to participate in our panel survey. We note that the views reflect the participant's personal experiences and does not reflect the view of the participant's employer. This material is based upon work supported by the Zuckerman STEM Leadership Program. This work has also been partially supported also by the PRIN 2017 project RexLearn (grant no. 2017TWNMH2), funded by the Italian Ministry of Education, University and Research; by the U.S.-Israel Energy Center managed by the Israel-U.S. Binational Industrial Research and Development (BIRD) Foundation; and by BMK, BMDW, and the Province of Upper Austria in the frame of the COMET Programme managed by FFG in the COMET Module S3AI.

References

[1] R. K. Knake, A cyberattack on the u.s. power grid, Tech. rep., Council on Foreign Relations (2017).
URL http://www.jstor.org/stable/resrep05652

[2] T. A. Mattei, Privacy, confidentiality, and security of health care information: Lessons from the recent wannacry cyberattack, World neurosurgery 104 (2017) 972—974. doi:10.1016/j.wneu.2017.06.104.
URL https://doi.org/10.1016/j.wneu.2017.06.104

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

[4] Y. Mirsky, T. Doitshman, Y. Elovici, A. Shabtai, Kitsune: An ensemble of autoencoders for online network intrusion detection, in: 25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018, The Internet Society, 2018.
URL http://wp.internetsociety.org/ndss/wp-content/uploads/sites/25/2018/02/ndss2018_03A-3_Mirsky_paper.pdf

[5] H. Liu, B. Lang, Machine learning and deep learning methods for intrusion detection systems: A survey, applied sciences 9 (20) (2019) 4396.

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

[7] Deepreflect: Discovering malicious functionality through binary reconstruction, in: 30th USENIX Security Symposium (USENIX Security 21), USENIX Association, 2021.
URL https://www.usenix.org/conference/usenixsecurity21/presentation/downing

[8] D. Ucci, L. Aniello, R. Baldoni, Survey of machine learning techniques for malware analysis, Computers & Security 81 (2019) 123–147.

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

[10] Y. Mirsky, W. Lee, The creation and detection of deepfakes: A survey, ACM Computing Surveys (CSUR) 54 (1) (2021) 1–41.

[11] 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).

[12] I. Zelinka, S. Das, L. Sikora, R. Šenkeřík, Swarm virus-next-generation virus and antivirus paradigm?, Swarm and Evolutionary Computation 43 (2018) 207–224.

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

[14] T. Gu, B. Dolan-Gavitt, 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.

[15] B. Biggio, F. Roli, Wild Patterns: Ten Years After the Rise of Adversarial Machine Learning, Pattern Recognition 84 (2018) 317–331.

[16] T. Orekondy, B. Schiele, M. Fritz, Knockoff Nets: Stealing Functionality of Black-Box Models, 2019, pp. 4954–4963.
URL https://openaccess.thecvf.com/content_CVPR_2019/html/Orekondy_Knockoff_Nets_Stealing_Functionality_of_Black-Box_Models_CVPR_2019_paper.html

[17] M. Fredrikson, S. Jha, T. Ristenpart, Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures, in: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, CCS '15, ACM, New York, NY, USA, 2015, pp. 1322–1333.

[18] T. Brewster, Fraudsters cloned company director's voice in $35 million bank heist, police find, https://www.forbes.com/sites/thomasbrewster/2021/10/14/huge-bank-fraud-uses-deep-fake-voice-tech-to-steal-millions/?sh=2e3ea2297559 (2021, note = (Accessed on 06/15/2022)).

[19] 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.

[20] M. Caldwell, J. Andrews, T. Tanay, L. Griffin, Ai-enabled future crime, Crime Science 9 (1) (2020) 1–13.

[21] R. R. Yager, Approximate reasoning as a basis for rule-based expert systems, IEEE Transactions on Systems, Man, and Cybernetics SMC-14 (4) (1984) 636–643. doi:10.1109/TSMC.1984.6313337.

[22] W. Zeng, R. L. Church, Finding shortest paths on real road networks: The case for a*, Int. J. Geogr. Inf. Sci. 23 (4) (2009) 531–543. doi:10.1080/13658810801949850.
URL https://doi.org/10.1080/13658810801949850

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

[24] Y. Wang, Q. Yao, J. T. Kwok, L. M. Ni, Generalizing from a few examples: A survey on few-shot learning, ACM Comput. Surv. 53 (3). doi:10.1145/3386252.
URL https://doi.org/10.1145/3386252

[25] M. T. Ribeiro, S. Singh, 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, ACM, New York, NY, USA, 2016, pp. 1135–1144.

[26] A. Castiglione, R. D. Prisco, A. D. Santis, U. Fiore, F. Palmieri, A botnet-based command and control approach relying on swarm intelligence, Journal of Network and Computer Applications 38 (2014) 22–33. doi:https://doi.org/10.1016/j.jnca.2013.05.002.
URL https://www.sciencedirect.com/science/article/pii/S1084804513001161

[27] J. A. Bland, M. D. Petty, T. S. Whitaker, K. P. Maxwell, W. A. Cantrell, Machine learning cyberattack and defense strategies, Computers & Security 92 (2020) 101738. doi:https://doi.org/10.1016/j.cose.2020.101738.
URL http://www.sciencedirect.com/science/article/pii/S0167404818309799

[28] Y. Mirsky, T. Mahler, I. Shelef, Y. Elovici, Ct-gan: Malicious tampering of 3d medical imagery using deep learning, in: 28th USENIX Security Symposium (USENIX Security 19), USENIX Association, Santa Clara, CA, 2019, pp. 461–478.
URL https://www.usenix.org/conference/usenixsecurity19/presentation/mirsky

[29] M. Schreyer, T. Sattarov, B. Reimer, D. Borth, Adversarial learning of deepfakes in accounting (2019). arXiv:1910.03810.

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

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

[32] C. Novo, 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, Association for Computing Machinery, New York, NY, USA, 2020, pp. 83–91. doi:10.1145/3411508.3421379.
URL https://doi.org/10.1145/3411508.3421379

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

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

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

[36] P. Marquardt, A. Verma, H. Carter, 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, 2011, pp. 551–562.

[37] Y. Abid, A. Imine, M. Rusinowitch, Sensitive attribute prediction for social networks users, in: EDBT/ICDT Workshops, 2018.

[38] G. Lin, S. Wen, Q.-L. Han, J. Zhang, Y. Xiang, Software vulnerability detection using deep neural networks: A survey, Proceedings of the IEEE 108 (10) (2020) 1825–1848.

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

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

[41] T. Rahman, M. Rochan, 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), 2019, pp. 1–8. doi:10.1109/AVSS.2019.8909869.

[42] X. Zhu, X. Jing, X. You, X. Zhang, T. Zhang, Video-based person re-identification by simultaneously learning intra-video and inter-video distance metrics, IEEE Transactions on Image Processing 27 (11) (2018) 5683–5695. doi:10.1109/TIP.2018.2861366.

[43] Y. Zhang, J. E. Meng, M. Pratama, Extractive document summarization based on convolutional neural networks, in: IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, 2016, pp. 918–922. doi:10.1109/IECON.2016.7793761.

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

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

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

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

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

[49] A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, 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, Curran Associates Inc., Red Hook, NY, USA, 2018, p. 6106–6116.

[50] H. Aghakhani, D. Meng, Y. Wang, C. Kruegel, G. Vigna, Bullseye polytope: A scalable clean-label poisoning attack with improved transferability, in: IEEE European Symposium on Security and Privacy, EuroS&P 2021, Vienna, Austria, September 6-10, 2021, IEEE, 2021, pp. 159–178. doi:10.1109/EuroSP51992.2021.00021.
URL https://doi.org/10.1109/EuroSP51992.2021.00021

[51] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, F. Roli, Evasion attacks against machine learning at test time, in: H. Blockeel, K. Kersting, S. Nijssen, F. Železný (Eds.), Machine

Learning and Knowledge Discovery in Databases (ECML PKDD), Part III, Vol. 8190 of LNCS, Springer Berlin Heidelberg, 2013, pp. 387–402.

[52] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks, in: International Conference on Learning Representations, 2014.
URL http://arxiv.org/abs/1312.6199

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

[54] M. Sharif, S. Bhagavatula, L. Bauer, 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, ACM, 2016, pp. 1528–1540.

[55] I. Shumailov, Y. Zhao, D. Bates, N. Papernot, R. D. Mullins, R. Anderson, Sponge examples: Energy-latency attacks on neural networks, in: IEEE European Symposium on Security and Privacy, EuroS&P 2021, Vienna, Austria, September 6-10, 2021, IEEE, 2021, pp. 212–231. doi:10.1109/EuroSP51992.2021.00024.
URL https://doi.org/10.1109/EuroSP51992.2021.00024

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

[57] A. Narayanan, V. Shmatikov, Robust de-anonymization of large sparse datasets, in: 2008 IEEE Symposium on Security and Privacy (sp 2008), 2008, pp. 111–125. doi:10.1109/SP.2008.33.

[58] S. Hidano, T. Murakami, S. Katsumata, S. Kiyomoto, 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), 2017, pp. 115–11509. doi:10.1109/PST.2017.00023.

[59] M. Juuti, S. Szyller, S. Marchal, N. Asokan, Prada: Protecting against dnn model stealing attacks, in: 2019 IEEE European Symposium on Security and Privacy (EuroS&P), 2019, pp. 512–527. doi:10.1109/EuroSP.2019.00044.

[60] H. Jia, C. A. Choquette-Choo, V. Chandrasekaran, N. Papernot, Entangled watermarks as a defense against model extraction (2021). arXiv:2002.12200.

[61] H. Zhang, H. Chen, Z. Song, D. S. Boning, I. S. Dhillon, 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.
URL https://openreview.net/forum?id=HylTBhA5tQ

[62] S. Woh, 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), 2018, pp. 89–92. doi:10.1109/SCIS-ISIS.2018.00025.

[63] E. Bagdasaryan, V. Shmatikov, Blind backdoors in deep learning models, in: 30th USENIX Security Symposium (USENIX Security 21), 2021, pp. 1505–1521.

[64] A. E. Cinà, A. Demontis, B. Biggio, F. Roli, M. Pelillo, Energy-Latency Attacks via Sponge Poisoning, arXiv:2203.08147 [cs]ArXiv: 2203.08147. URL http://arxiv.org/abs/2203.08147

[65] Y. Yao, H. Li, H. Zheng, 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, Association for Computing Machinery, New York, NY, USA, 2019, p. 2041–2055. doi:10.1145/3319535.3354209.
URL https://doi.org/10.1145/3319535.3354209

[66] S. Wang, S. Nepal, C. Rudolph, M. Grobler, S. Chen, T. Chen, Backdoor attacks against transfer learning with pre-trained deep learning models, IEEE Transactions on Services Computing (2020) 1–1doi:10.1109/TSC.2020.3000900.

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

[68] A. Demontis, M. Melis, M. Pintor, M. Jagielski, B. Biggio, A. Oprea, C. Nita-Rotaru, 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.

[69] M. Pintor, L. Demetrio, A. Sotgiu, M. Melis, A. Demontis, B. Biggio, secml: Secure and explainable machine learning in python, SoftwareX 18 (2022) 101095.

[70] M.-I. Nicolae, M. Sinn, M. N. Tran, B. Buesser, A. Rawat, M. Wistuba, V. Zantedeschi, N. Baracaldo, B. Chen, H. Ludwig, I. Molloy, B. Edwards, Adversarial robustness toolbox v1.2.0, CoRR 1807.01069.
URL https://arxiv.org/pdf/1807.01069

[71] 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, R. Long, Technical report on the cleverhans v2.1.0 adversarial examples library, arXiv preprint arXiv:1610.00768.

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

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

[74] S. Singh, 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), IEEE, 2020, pp. 1074–1079.

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

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

[77] M. Matta, G. C. Cardarilli, L. Di Nunzio, R. Fazzolari, D. Giardino, M. re, F. Silvestri, S. Spanò, Q-rts: a real-time swarm intelligence based on multi-agent q-learning, Electronics Lettersdoi:10.1049/el.2019.0244.

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

[79] S. Calzavara, G. Tolomei, A. Casini, M. Bugliesi, S. Orlando, A supervised learning approach to protect client authentication on the web, ACM Trans. Web 9 (3). doi:10.1145/2754933.
URL https://doi.org/10.1145/2754933

[80] K. Horák, B. Bošanský, P. Tomášek, C. Kiekintveld, C. Kamhoua, Optimizing honeypot strategies against dynamic lateral movement using partially observable stochastic games, Computers & Security 87 (2019) 101579. doi:https://doi.org/10.1016/j.cose.2019.101579.
URL http://www.sciencedirect.com/science/article/pii/S0167404819300665

[81] A. White, A. Matthews, K. Snow, F. Monrose, Phonotactic reconstruction of encrypted voip conversations: Hookt on fon-iks, 2011, pp. 3 – 18. doi:10.1109/SP.2011.34.

[82] L. Sidi, A. Nadler, A. Shabtai, Maskdga: An evasion attack against dga classifiers and adversarial defenses, IEEE Access 8 (2020) 161580–161592.

[83] S. Datta, Deepobfuscode: Source code obfuscation through sequence-to-sequence networks (2020). arXiv:1909.01837.

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

[85] B. I. Messaoud, K. Guennoun, M. Wahbi, 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), IEEE, 2016, pp. 1–6.

[86] J. Chen, C. Su, K.-H. Yeh, M. Yung, Special issue on advanced persistent threat (2018).

[87] A. Alshamrani, S. Myneni, A. Chowdhary, D. Huang, A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities, IEEE Communications Surveys & Tutorials 21 (2) (2019) 1851–1877.

[88] M. Goldblum, A. Schwarzschild, A. Patel, T. Goldstein, Adversarial attacks on machine learning systems for high-frequency trading, in: Proceedings of the Second ACM International Conference on AI in Finance, 2021, pp. 1–9.

[89] Y. Chen, Y. Tan, B. Zhang, Exploiting vulnerabilities of load forecasting through adversarial attacks, in: Proceedings of the Tenth ACM International Conference on Future Energy Systems, 2019, pp. 1–11.

[90] 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, (Accessed on 04/15/2021) (2019).

[91] Z. Li, D. Zou, S. Xu, X. Ou, H. Jin, S. Wang, Z. Deng, Y. Zhong, Vuldeepecker: A deep learning-based system for vulnerability detection, Proceedings 2018 Network and Distributed System Security Symposiumdoi:10.14722/ndss.2018.23158.
URL http://dx.doi.org/10.14722/ndss.2018.23158

[92] Z. Li, D. Zou, S. Xu, Z. Chen, Y. Zhu, H. Jin, Vuldeelocator: a deep learning-based fine-grained vulnerability detector, IEEE Transactions on Dependable and Secure Computing.

[93] X. Xu, C. Liu, Q. Feng, H. Yin, L. Song, D. Song, Neural network-based graph embedding for cross-platform binary code similarity detection, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 2017, pp. 363–376.

[94] Y. Wang, P. Jia, L. Liu, C. Huang, Z. Liu, A systematic review of fuzzing based on machine learning techniques, PloS one 15 (8) (2020) e0237749.

[95] C. Lavaud, R. Gerzaguet, M. Gautier, O. Berder, E. Nogues, S. Molton, Whispering devices: A survey on how side-channels lead to compromised information, Journal of Hardware and Systems Security 5 (2) (2021) 143–168.

[96] M. H. Abd El-Jawad, R. Hodhod, Y. M. K. Omar, Sentiment analysis of social media networks using machine learning, in: 2018 14th International Computer Engineering Conference (ICENCO), 2018, pp. 174–176. doi:10.1109/ICENCO.2018.8636124.

[97] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, 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, 2015, pp. 142–154.

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

[99] J. V. Monaco, What are you searching for? a remote keylogging attack on search engine autocomplete, in: 28th USENIX Security Symposium (USENIX Security 19), USENIX Association, Santa Clara, CA, 2019, pp. 959–976.
URL https://www.usenix.org/conference/usenixsecurity19/presentation/monaco

[100] A. Malhotra, L. Totti, W. Meira Jr., P. Kumaraguru, 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, IEEE Computer Society, USA, 2012, p. 1065–1070. doi:10.1109/ASONAM.2012.184.
URL https://doi.org/10.1109/ASONAM.2012.184

[101] N. Haim, G. Vardi, G. Yehudai, O. Shamir, M. Irani, Reconstructing training data from trained neural networks, arXiv preprint arXiv:2206.07758.

[102] S. Alrabaee, K.-K. R. Choo, M. Qbea'h, M. Khasawneh, Bindeep: Binary to source code matching using deep learning, in: 2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), IEEE, 2021, pp. 1100–1107.

[103] J. Li, L. Zhou, H. Li, L. Yan, H. Zhu, Dynamic Traffic Feature Camouflaging via Generative Adversarial Networks, in: 2019 IEEE Conference on Communications and Network Security (CNS), 2019, pp. 268–276. doi:10.1109/CNS.2019.8802772.

[104] Our work with the dnc: Setting the record straight (Jun 2020).
URL https://www.crowdstrike.com/blog/bears-midst-intrusion-democratic-national-committee/

[105] D. Lunghi, J. Horejsi, C. Pernet, Untangling the patchwork cyberespionage group (2017).
URL https://documents.trendmicro.com/assets/tech-brief-untangling-the-patchwork-cyberespionage-group.pdf

[106] R. Leong, D. Perez, T. Dean, Messagetap: Who's reading your text messages? (2019).
URL https://www.fireeye.com/blog/threat-research/2019/10/messagetap-who-is-reading-your-text-messages.html

[107] E. Brumaghin, H. Unterbrink, 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).

[108] L. Arsene, Oil & gas spearphishing campaigns drop agent tesla spyware in advance of historic opec+ deal (2020).
URL https://labs.bitdefender.com/2020/04/oil-gas-spearphishing-campaigns-drop-agent-tesla-spyware-in-advance-of-historic-ope

[109] X. Zhang, Analysis of new agent tesla spyware variant (2018).
URL https://www.fortinet.com/blog/threat-research/analysis-of-new-agent-tesla-spyware-variant.html

[110] R. Mueller, Indictment - united states of america vs. viktor borisovich netyksho, et al. (2018).
URL https://www.justice.gov/file/1080281/download

[111] G. Beni, Swarm intelligence, Complex Social and Behavioral Systems: Game Theory and Agent-Based Models (2020) 791–818.

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

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

[114] M. Yousefi, N. Mtetwa, Y. Zhang, 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), 2018, pp. 212–217. doi:10.1109/TrustCom/BigDataSE.2018.00041.

[115] R. Wu, J. Gong, W. Tong, B. Fan, Network attack path selection and evaluation based on q-learning, Applied Sciences 11 (1). doi:10.3390/app11010285.
URL https://www.mdpi.com/2076-3417/11/1/285

[116] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, z. Chen, P. Nguyen, R. Pang, I. Lopez Moreno, Y. Wu, Transfer learning from speaker verification to multispeaker text-to-speech synthesis, in: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), Advances in Neural Information Processing Systems 31, Curran Associates, Inc., 2018, pp. 4480–4490.
URL http://papers.nips.cc/paper/7700-transfer-learning-from-speaker-verification-to-mu
pdf

[117] N. O. Leslie, R. E. Harang, L. P. Knachel, A. Kott, Statistical models for the number of successful cyber intrusions, CoRR abs/1901.04531. arXiv:1901.04531.
URL http://arxiv.org/abs/1901.04531

[118] A. Kumar, A. Biswas, S. Sanyal, ecommercegan : A generative adversarial network for e-commerce (2018). arXiv:1801.03244.

[119] M. Manning, G. T. Wong, T. Graham, T. Ranbaduge, P. Christen, K. Taylor, R. Wortley, T. Makkai, P. Skorich, Towards a 'smart'cost–benefit tool: using machine learning to predict the costs of criminal justice policy interventions, Crime Science 7 (1) (2018) 12.

[120] A. Fuller, Z. Fan, C. Day, C. Barlow, Digital twin: Enabling technologies, challenges and open research, IEEE Access 8 (2020) 108952–108971.

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

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

[123] Z. Fang, J. Wang, B. Li, S. Wu, Y. Zhou, H. Huang, Evading anti-malware engines with deep reinforcement learning, IEEE Access 7 (2019) 48867–48879. doi:10.1109/ACCESS.2019.2908033.

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

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

[126] P. Mozur, Looking through the eyes of china's surveillance state, accessed: June 2018 (2018).
URL https://www.nytimes.com/2018/07/16/technology/china-surveillance-state.html

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

[128] C. Ding, K. Huang, V. M. Patel, B. C. Lovell, Special issue on video surveillance-oriented biometrics, Pattern Recognition Letters 107 (2018) 1–2.

[129] P. Bontrager, A. Roy, J. Togelius, N. Memon, 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), IEEE, 2018, pp. 1–9.

[130] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. W. D. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee, L. Juvela, P. Alku, Y. Peng, H. Hwang, Y. Tsao, H. Wang, S. L. Maguer, M. Becker, Z. Ling, Asvspoof 2019: A large-scale public database of synthesized,

converted and replayed speech, Comput. Speech Lang. 64 (2020) 101114. doi:10.1016/j.csl.2020.101114.
URL https://doi.org/10.1016/j.csl.2020.101114

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

[132] A. Maiti, M. Jadliwala, J. He, I. Bilogrevic, Side-channel inference attacks on mobile keypads using smartwatches, IEEE Transactions on Mobile Computing 17 (9) (2018) 2180–2194.

[133] A. Compagno, M. Conti, D. Lain, 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, 2017, pp. 703–715.

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

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

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

[137] Y. Wang, W. Cai, T. Gu, W. Shao, I. Khalil, 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, 2018, pp. 254–263.

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

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

[140] 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, Springer, 2018, pp. 263–280.

[141] J. Lim, T. Price, F. Monrose, J. Frahm, Revisiting the threat space for vision-based keystroke inference attacks, in: A. Bartoli, A. Fusiello (Eds.), Computer Vision - ECCV 2020 Workshops - Glasgow, UK, August 23-28, 2020, Proceedings, Part V, Vol. 12539 of Lecture Notes in Computer Science, Springer, 2020, pp. 449–461.

doi:10.1007/978-3-030-68238-5_33.
URL https://doi.org/10.1007/978-3-030-68238-5_33

[142] S. Nam, S. Jeon, H. Kim, J. Moon, Recurrent gans password cracker for iot password security enhancement, Sensors 20 (11) (2020) 3106.

[143] J. Seymour, P. Tully, Generative models for spear phishing posts on social media, arXiv preprint arXiv:1802.05196.

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

[145] L. Lerman, G. Bontempi, O. Markowitch, Power analysis attack: an approach based on machine learning, International Journal of Applied Cryptography 3 (2) (2014) 97–115.

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

[147] D. Brumley, D. Boneh, Remote timing attacks are practical, Computer Networks 48 (5) (2005) 701–716.

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

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

[150] S. Picek, A. Heuser, A. Jovic, S. Bhasin, 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 2019 (1) (2019) 1–29.

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

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

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

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

[155] G. Perin, Ł. Chmielewski, L. Batina, S. Picek, Keep it unsupervised: Horizontal attacks meet deep learning, IACR Transactions on Cryptographic Hardware and Embedded Systems (2021) 343–372.

[156] D. X. Song, D. A. Wagner, X. Tian, Timing analysis of keystrokes and timing attacks on ssh., in: USENIX Security Symposium, Vol. 2001, 2001.

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

[158] X. Xu, C. Liu, Q. Feng, H. Yin, L. Song, 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 Securitydoi:10.1145/3133956.3134018.
URL http://dx.doi.org/10.1145/3133956.3134018

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

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

[161] S. H. Ding, B. C. Fung, 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), IEEE, 2019, pp. 472–489.

[162] Y. Duan, X. Li, J. Wang, 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.

[163] 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.

[164] S. Yun, M. Jeong, R. Kim, J. Kang, H. J. Kim, Graph transformer networks, in: H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, 2019, pp. 11960–11970.
URL https://proceedings.neurips.cc/paper/2019/hash/9d63484abb477c97640154d40595a3bb-Abstract.html

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

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

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

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

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

[170] Z. Li, D. Zou, J. Tang, Z. Zhang, M. Sun, H. Jin, A comparative study of deep learning-based vulnerability detection system, IEEE Access 7 (2019) 103184–103197.

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

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

[173] D. She, R. Krishna, L. Yan, S. Jana, B. Ray, Mtfuzz: fuzzing with a multi-task neural network, in: P. Devanbu, M. B. Cohen, T. Zimmermann (Eds.), ESEC/FSE '20: 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, Virtual Event, USA, November 8-13, 2020, ACM, 2020, pp. 737–749. doi:10.1145/3368089.3409723.
URL https://doi.org/10.1145/3368089.3409723

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

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

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

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

[178] H. Samulowitz, R. Memisevic, Learning to solve qbf, in: AAAI, Vol. 7, 2007, pp. 255–260.

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

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

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

[182] Y. Ghazi, Z. Anwar, R. Mumtaz, S. Saleem, 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), IEEE, 2018, pp. 129–134.

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

[184] L. Akoglu, H. Tong, D. Koutra, Graph based anomaly detection and description: a survey, Data mining and knowledge discovery 29 (3) (2015) 626–688.

[185] J. Schwartz, H. Kurniawati, Autonomous penetration testing using reinforcement learning, arXiv preprint arXiv:1905.05965.

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

[187] R. Dabre, C. Chu, A. Kunchukuttan, A survey of multilingual neural machine translation, ACM Computing Surveys (CSUR) 53 (5) (2020) 1–38.

[188] Z. Nasar, S. W. Jaffry, M. K. Malik, Textual keyword extraction and summarization: State-of-the-art, Information Processing & Management 56 (6) (2019) 102088.

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

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

[191] 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).

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

[193] L. Batina, S. Bhasin, D. Jap, S. Picek, CSI NN: Reverse engineering of neural network architectures through electromagnetic side channel, in: 28th USENIX Security Symposium (USENIX Security 19), USENIX Association, Santa Clara, CA, 2019, pp. 515–532.
URL https://www.usenix.org/conference/usenixsecurity19/presentation/batina

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

[195] B. Zhou, M. Elbadry, R. Gao, 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, 2017, pp. 42–55.

[196] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, R. Qu, A survey of deep learning-based object detection, IEEE Access 7 (2019) 128837–128868.

[197] Y. Ren, X. Tan, T. Qin, S. Zhao, Z. Zhao, T. Liu, Almost unsupervised text to speech and automatic speech recognition, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, Vol. 97 of Proceedings of Machine Learning Research, PMLR, 2019, pp. 5410–5419.
URL http://proceedings.mlr.press/v97/ren19a.html

[198] A. Al-Hababi, 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), IEEE, 2020, pp. 1–5.

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

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

[201] 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 59 (4) (2008) 662–674.

[202] K. Hao, Deepfakes may be a useful tool for spies — mit technology review, https://www.technologyreview.com/2019/06/14/134934/deepfakes-spies-espionage/, (Accessed on 06/21/2022) (2019).

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

[204] J. Salminen, R. G. Rao, S.-g. Jung, S. A. Chowdhury, 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, Springer, 2020, pp. 101–120.

[205] D. Spiliotopoulos, D. Margaris, C. Vassilakis, Data-assisted persona construction using social media data, Big Data and Cognitive Computing 4 (3) (2020) 21.

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

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

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

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

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

[211] R. Kong, X. Tong, Dynamic weighted heuristic trust path search algorithm, IEEE Access 8 (2020) 157382–157390.

[212] S. Roller, E. Dinan, N. Goyal, D. Ju, M. Williamson, Y. Liu, J. Xu, M. Ott, E. M. Smith, Y. Boureau, J. Weston, Recipes for building an open-domain chatbot, in: P. Merlo, J. Tiedemann, R. Tsarfaty (Eds.), Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021, Association for Computational Linguistics,

2021, pp. 300–325. doi:10.18653/v1/2021.eacl-main.24.
URL https://doi.org/10.18653/v1/2021.eacl-main.24

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

[214] J. Seymour, P. Tully, Weaponizing data science for social engineering: Automated e2e spear phishing on twitter, Black Hat USA 37 (2016) 1–39.

[215] A. Das, R. Verma, Automated email generation for targeted attacks using natural language (2019). arXiv:1908.06893.

[216] Z. Fu, X. Tan, N. Peng, D. Zhao, R. Yan, Style transfer in text: Exploration and evaluation, in: S. A. McIlraith, K. Q. Weinberger (Eds.), Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, AAAI Press, 2018, pp. 663–670.
URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17015

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

[218] R. Bitton, K. Boymgold, R. Puzis, A. Shabtai, Evaluating the information security awareness of smartphone users, in: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 2020, pp. 1–13.

[219] A. Solomon, M. Michaelshvili, R. Bitton, B. Shapira, L. Rokach, R. Puzis, A. Shabtai, Contextual security awareness: A context-based approach for assessing the security awareness of users, Knowledge-Based Systems 246 (2022) 108709.

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

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

[222] M. Ghiassi, S. Lee, A domain transferable lexicon set for twitter sentiment analysis using a supervised machine learning approach, Expert Systems with Applications 106 (2018) 197–216.

[223] M. Rathi, A. Malik, D. Varshney, R. Sharma, S. Mendiratta, Sentiment analysis of tweets using machine learning approach, in: 2018 Eleventh International Conference on Contemporary Computing (IC3), 2018, pp. 1–3. doi:10.1109/IC3.2018.8530517.

[224] Black Hat USA 2018.
URL https://www.blackhat.com/us-18/arsenal.html#social-mapper-social-media-correlation-through-facial-recognition

[225] H. Pellet, S. Shiaeles, S. Stavrou, Localising social network users and profiling their movement, Computers & Security 81 (2019) 49–57.

[226] W. Zhang, R. Lau, S. Liao, R.-W. Kwok, A probabilistic generative model for latent business networks mining, International Conference on Information Systems, ICIS 2012 2 (2012) 1102–1118.

[227] Z. Ma, O. Sheng, G. Pant, Discovering company revenue relations from news: A network approach, Decision Support Systems 47 (2009) 408–414. doi:10.1016/j.dss.2009.04.007.

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

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

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

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

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

[233] J. Breier, X. Hou, D. Jap, L. Ma, S. Bhasin, Y. Liu, Practical fault attack on deep neural networks, in: D. Lie, M. Mannan, M. Backes, X. Wang (Eds.), Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS 2018, Toronto, ON, Canada, October 15-19, 2018, ACM, 2018, pp. 2204–2206. doi:10.1145/3243734.3278519.
URL https://doi.org/10.1145/3243734.3278519

[234] S. Li, S. Ma, M. Xue, B. Z. H. Zhao, Deep learning backdoors, in: Security and Artificial Intelligence, 2022, pp. 313–334. doi:10.1007/978-3-030-98795-4_13.
URL https://doi.org/10.1007/978-3-030-98795-4_13

[235] K. Hasegawa, M. Yanagisawa, N. Togawa, Trojan-net classification for gate-level hardware design utilizing boundary net structures, IEICE TRANSACTIONS on Information and Systems 103 (7) (2020) 1618–1622.

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

[237] A. Demontis, M. Melis, M. Pintor, M. Jagielski, B. Biggio, A. Oprea, C. Nita-Rotaru, 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.

[238] D. Maiorca, A. Demontis, B. Biggio, F. Roli, G. Giacinto, Adversarial Detection of Flash Malware: Limitations and Open Issues, Computers & Security 96 (2020) 101901. doi:https://doi.org/10.1016/j.cose.2020.101901.
URL http://www.sciencedirect.com/science/article/pii/S0167404820301760

[239] K. K. Ispoglou, M. Payer, malwash: Washing malware to evade dynamic analysis, in: 10th USENIX Workshop on Offensive Technologies (WOOT 16), USENIX Association, Austin, TX, 2016.
URL https://www.usenix.org/conference/woot16/workshop-program/presentation/ispoglou

[240] O. Suciu, S. E. Coull, J. Johns, Exploring adversarial examples in malware detection, in: 2019 IEEE Security and Privacy Workshops (SPW), 2019, pp. 8–14. doi:10.1109/SPW.2019.00015.

[241] L. Demetrio, B. Biggio, G. Lagorio, F. Roli, A. Armando, Functionality-preserving black-box optimization of adversarial windows malware, IEEE Trans. Inf. Forensics Secur. 16 (2021) 3469–3478. doi:10.1109/TIFS.2021.3082330.
URL https://doi.org/10.1109/TIFS.2021.3082330

[242] F. Pierazzi, F. Pendlebury, J. Cortellazzi, L. Cavallaro, Intriguing properties of adversarial ml attacks in the problem space, in: 2020 IEEE Symposium on Security and Privacy (SP), 2020, pp. 1332–1349. doi:10.1109/SP40000.2020.00073.

[243] H. S. Anderson, A. Kharkar, B. Filar, D. Evans, P. Roth, Learning to evade static pe machine learning malware models via reinforcement learning (2018). arXiv:1801.08917.

[244] F. Zhiyang, J. Wang, B. Li, S. Wu, Y. Zhou, H. Huang, Evading anti-malware engines with deep reinforcement learning, IEEE Access PP (2019) 1–1. doi:10.1109/ACCESS.2019.2908033.

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

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

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

[248] A. G. Sutro, Machine-Learning Based Evaluation of Access Control Lists to Identify Anomalies (Jan. 2020).
URL https://www.tdcommons.org/dpubs_series/2870

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

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

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

[252] J. Gao, J. Lanchantin, M. L. Soffa, Y. Qi, Black-box generation of adversarial text sequences to evade deep learning classifiers, in: 2018 IEEE Security and Privacy Workshops (SPW), 2018, pp. 50–56. doi:10.1109/SPW.2018.00016.

[253] Y. Li, Y. Wang, Y. Wang, L. Ke, Y.-a. Tan, A feature-vector generative adversarial network for evading pdf malware classifiers, Information Sciences 523 (2020) 38–48.

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

[255] M. Rigaki, S. Garcia, Bringing a gan to a knife-fight: Adapting malware communication to avoid detection, in: 2018 IEEE Security and Privacy Workshops (SPW), 2018, pp. 70–75. doi:10.1109/SPW.2018.00019.

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

[257] M. Abadi, D. G. Andersen, Learning to Protect Communications with Adversarial Neural Cryptography, arXiv.
URL https://arxiv.org/abs/1610.06918

[258] M. Guri, Y. Elovici, Bridgeware: The air-gap malware, Communications of the ACM 61 (4) (2018) 74–82.

[259] S. Jiang, D. Ye, J. Huang, Y. Shang, Z. Zheng, Smartsteganogaphy: Lightweight generative audio steganography model for smart embedding application, Journal of Network and Computer Applications 165 (2020) 102689.

[260] Fraudsters cloned company director's voice in $35 million bank heist, police find, https://www.forbes.com/sites/thomasbrewster/2021/10/14/huge-bank-fraud-uses-deep-fake-voice-tech-to-steal-millions/?sh=23254d367559, (Accessed on 08/17/2022) (2020).

[261] A. Navalny, European mps targeted by deepfake video calls imitating russian opposition — russia — the guardian, https://www.theguardian.com/world/2021/apr/22/european-mps-targeted-by-deepfake-video-calls-imitating-russian-opposition, (Accessed on 08/17/2022) (2021).

[262] FBI, Fbi: Scammers are interviewing for remote jobs using deepfake tech — mashable, https://mashable.com/article/deepfake-job-interviews-fbi#:~:text=Deepfakes%20involve%20using%20AI%2Dpowered,say%20whatever%20you'd%20like., (Accessed on 08/17/2022) (2022).

[263] J. Vincent, Binance executive claims scammers made a deepfake of him - the verge, https://www.theverge.com/2022/8/23/23318053/binance-comms-crypto-chief-deepfake-scam-claim-patrick-hillmann, (Accessed on 09/07/2022) (2022).

[264] E. Hollnagel, D. D. Woods, N. Leveson, Resilience engineering: Concepts and precepts, Ashgate Publishing, Ltd., 2006.

[265] D. D. Woods, E. Hollnagel, Joint cognitive systems: Patterns in cognitive systems engineering, CRC Press, 2006.

[266] C. P. Software, 2021 cyber security report - check point software, https://www.checkpoint.com/pages/cyber-security-report-2021/, (Accessed on 06/23/2022) (2021).

[267] A. Kuppa, N.-A. Le-Khac, Black box attacks on explainable artificial intelligence(xai) methods in cyber security, in: 2020 International Joint Conference on Neural Networks (IJCNN), 2020, pp. 1–8. doi:10.1109/IJCNN48605.2020.9206780.

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

[269] 2019 data breach investigations report, Verizon, Inc.

[270] Final report - national security commission on artificial intelligence, National Security Commission on Artificial Intelligence.

[271] The emergence of offensive ai, Forrester.

[272] Preparing for ai-enabled cyberattacks, MIT Technology Review Insights.

8. Vitae

Yisroel Mirsky (Ph.D. 2018) is a tenure-track lecturer and Zuckerman Faculty Scholar in the Department of Software and Information Systems Engineering at Ben-Gurion University. He is also the head of the Offensive AI Research Lab at BGU. His main research interests include deepfakes, adversarial machine learning, anomaly detection, and intrusion detection. Dr. Mirsky has published his work in some of the best security venues: USENIX, CCS, NDSS, Euro S&P, Black Hat, DEF CON, RSA, CSF, AISec, etc. His research has also been featured in many well-known media outlets: Popular Science, Scientific American, Wired, The Wall Street Journal, Forbes, and BBC. Some of his works, include the exposure of vulnerabilities in the US 911 emergency services and research into the threat of deepfakes in medical scans, both featured in The Washington Post.

Ambra Demontis is an Assistant Professor at the University of Cagliari, Italy. She received her M.Sc. degree (Hons.) in Computer Science and her Ph.D. degree in Electronic Engineering and Computer Science from the University of Cagliari, Italy, respectively, in 2014 and 2018. Her research interests include secure machine learning, kernel methods, and computer security. She co-organizes the AISec workshop, serves on the program committee of different conferences and workshops, such as IJCAI and DLS, and as a reviewer for several journals, such as TNNLS, TOPS, Machine Learning, and Pattern Recognition. She is a Member of the IEEE and the IAPR.

Jaidip Kotak is a Doctorate student in the Information Systems Engineering department at Ben-Gurion University of the Negev (BGU), Israel. His primary areas of interest are computer and network security, general hacking, and application of machine learning in detecting cyber attacks. Jaidip has a M.Tech. with a specialization in Cyber Security & Incident Response and B.E. in Information Technology from India.

Ram Shankar is a data cowboy within the Azure Security Data Science team at Microsoft where he works at the intersection of machine learning and security. He is the founder of the Security Data Science Colloquium — the only avenue where security data scientists from every major cloud provider congregate. Ram is also an affiliate at the Berkman Klein Center at Harvard University.

Deng Gelei is a 2nd year Ph.D. student at the School of Computer Science and Engineering, Nanyang Technological University. He received his bachelor's degree from Singapore University of Technology and Design. Before Ph.D. study, he worked as research engineer and penetration tester at Singapore Agency for Science, Technology and Research (A*STAR). His research interests include software security and system security. He has recently been focusing on uncovering vulnerabilities in complex cyber-physical systems through software engineering techniques.

Liu Yang graduated in 2005 with a Bachelor of Computing (Honours) in National University of Singapore (NUS). In 2010, he obtained his Ph.D. and started his post-doctoral work in NUS and MIT. In 2012, he joined Nanyang Technological University (NTU), and currently is a full professor and Director of the cybersecurity lab in NTU. Dr. Liu specializes in software engineering, cybersecurity and artificial intelligence. His research has bridged the gap between the theory and practical usage of program analysis, data analysis and AI to evaluate the design and implementation of software for high assurance and security. By now, he has more than 400 publications in top tier conferences and journals. He has received a number of prestigious awards including MSRA Fellowship, TRF Fellowship, Nanyang Assistant Professor, Tan Chin Tuan Fellowship, Nanyang Research Award 2019, ACM Distinguished Speaker, NRF Investigatorship, and 20 best paper awards and one most influence system award in top software engineering conferences like ASE, FSE and ICSE

Xiangyu Zhang has substantial research experience in the areas of program analysis, security and AI. His group has produced a list of binary analysis tools that can identify and extract functional components from binaries, reuse binary functional components for rendering forensic evidence in memory, rewrite x86 binaries without relocation information, lift binary execution traces to stand-alone programs that can be compiled and executed on other platforms, vet x86 real world binaries and thousands of iOS apps, and stealthily monitor binary execution. Zhang's group has developed novel attack forensics techniques that instrument applications such that a minimal set of critical application events can be recorded in addition to system level events to facilitate producing comprehensive cyber-attack provenance. Zhang has also substantial experience in AI systems and AI security, especially in AI backdoor attack detection and mitigation.

Maura Pintor is a Postdoctoral Researcher at the PRA Lab, in the Department of Electrical and Electronic Engineering of the University of Cagliari, Italy. She received her PhD in Electronic and Computer Engineering from the University of Cagliari in 2022. Her research focuses on machine learning security.

Wenke Lee is a Professor and John P. Imlay Jr. Chair in the School of Cybersecurity and Privacy in the College of Computing at The Georgia Institute of Technology. He was also the Director of the Institute for Information Security & Privacy (IISP) at Georgia Tech from 2015 to 2021. He received his Ph.D. in Computer Science from Columbia University in 1999. His research interests include systems and network security, machine learning, and applied cryptography, and he has been the PI of several large-scale projects funded by the government and industry. He is an ACM Fellow and IEEE Fellow.

Yuval Elovici is the director of the Telekom Innovation Laboratories at Ben-Gurion University of the Negev (BGU), head of BGU's Cyber Security

Research Center, and a professor in the Department of Software and Information Systems Engineering at BGU. He holds B.Sc. and M.Sc. degrees in computer and electrical engineering from BGU and a Ph.D. in information systems from Tel Aviv University. His primary research interests are computer and network security, cyber security, web intelligence, information warfare, side-channel attacks, AI security, and machine learning. Prof. Elovici also consults professionally in the area of cyber security, sharing his expertise with both startups and large international companies, and is the co-founder of Morphisec, a startup that develops innovative cyber security mechanisms related to moving target defense, and CyberMed which focuses on securing medical imaging devices.

Battista Biggio (MSc 2006, PhD 2010) is Assistant Professor at the University of Cagliari, Italy, and co-founder of the cybersecurity company Pluribus One. He has provided pioneering contributions in machine-learning security, playing a leading role in this field. He has managed six research projects, and regularly serves as a PC member for ICML, NeurIPS, ICLR, IEEE Symp. S&P, and USENIX Security. He chaired IAPR TC1 (2016-2020), co-organized S+SSPR, AISec and DLS, and served as Associate Editor for IEEE TNNLS, IEEE CIM, and Pattern Recognition. He is a senior member of IEEE and ACM, and a member of IAPR and ELLIS.

Appendix A. The Complete Questionnaire

Appendix A.1. Rating the Threat

In an attack on an organization, there are 7 malicious activities that can be enhanced using AI: automation, information gathering, campaign resilience, credential theft, social engineering, stealth, and exploit development.

Please rate accordingly:

Harm. How harmful is an attacker with AI in this task?
(damage, attack persistence, evasion, defense effort)

Profit. How beneficial is AI to the attacker in this task? (compared to using non-AI methods)
(attack success, flexibility, coverage, automation, and persistence). Assume that the AI tool has already been implemented.

Achievability How easy is it for the attacker to use AI for this task?
(implement, train, and deploy the AI)

Defeatability How easy is it for the defender to detect or prevent it?
(1=hard to defeat, 7=easy to defeat)

Activity: Automation.
Attack Customization (e.g., adjusting an exploit) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __
Coordinated Attacks: (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Information Sharing (among bots or threat agents) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Next-hop Targeting (e.g., lateral movement) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Phishing Campaigns (e.g., automated into collection crafting of spear phishing emails, calls, ...)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Point of Entry Detection (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Activity: Information Gathering (IG).
Mining OSINT (e.g., parsing websites, retrieving relevant info, ...) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

AI Model Theft (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Spying (e.g., collecting and mining conversations from the microphone, locations from the camera,...)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Activity: Campaign Resilience (CR).
Malware Obfuscation (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Persistent Backdoors (e.g., automated reinfection, backdoor info shared among bots, ...)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Virtualization Detection (anti-forensics for malware) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Activity: Credential Theft (CT).
Biometric Spoofing (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Device Cache Mining (1 = low, 7 = high)

Harm: __, Profit: __, Achievability: __, Defeatability: __

Implicit Key Logging (e.g., using smartphone acceleration, keystroke sounds, ...)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Intelligent Password Guessing (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Side Channel Mining (e.g., memory or timing patterns) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Activity: Social Engineering (SE).
Impersonation (e.g., voice, text, video deepfakes and online social profiles) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Persona Building (e.g., a targeted trustworthy/attractive online profile) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Spear Phishing (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Target Selection (e.g., weakest link with asset) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Activity Tracking (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Activity: Stealth.
Covering Tracks (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Web Domain Name Generation (e.g., DGAs to avoid detection and blacklisting)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Evading Network or Host-based Intrusion Detection Systems (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Evading Insider Detection Systems (e.g., replicate access pattern of other user)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Evading Email Filter (i.e., for SPAM and phishing) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Data Exfiltration (e.g., evading firewall or over an air-gap for an isolated network)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Propagation (lateral movement over a network) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Scanning (e.g., local host, network assets, ports, vulnerabilities, ...) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Activity: Exploit Development (ED).
Reverse Engineering (i.e., to assist in manually finding a vulnerability or steal IP) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Vulnerability Detection (e.g., intelligent fuzzing, static analysis, ...) (1 = low, 7 = high)
Harm: __, Profit: __, Achievability: __, Defeatability: __

Appendix A.2. The Impact on the Cyber Kill Chain

In an advanced persistent threat (APT) an adversary follows 14 tactics to attack an organization according to the MITRE A&TTACK matrix. However, at each step the defender can stop the attack and effectively kill the chain of events, preventing the attacker from reaching its goal.

Compared to using conventional methods, AI helps the attacker in...
(strongly disagree, disagree somewhat disagree, neutral, somewhat agree, agree, strongly agree)
(1) Reconnaissance: __, (2) Resource Development: __, (3) Initial Access: __, (4) Execution: __, (5)Persistence: __, (6) Privilege Escalation: __, (7) Defense Evasion: __, (8) Credential Access: __, (9) Discovery: __, (10) Lateral Movement: __, (11) Collection: __, (12) Command & Control: __, (13) Exfiltration: __, (14) Impact: __

For each tactic, would AI help the attacker more than the defender?
(strongly disagree, disagree somewhat disagree, neutral, somewhat agree, agree, strongly agree)
(1) Reconnaissance: __, (2) Resource Development: __, (3) Initial Access: __, (4) Execution: __, (5)Persistence: __, (6) Privilege Escalation: __, (7) Defense Evasion: __, (8) Credential Access: __, (9) Discovery: __, (10) Lateral Movement: __, (11) Collection: __, (12) Command & Control: __, (13) Exfiltration: __, (14) Impact: __