Carnegie Mellon University
Software Engineering Institute

COMMENTS ON NISTIR 8269 (A TAXONOMY AND TERMINOLOGY OF ADVERSARIAL MACHINE LEARNING)

Dr. April Galyardt, Dr. Jonathan M. Spring, Dr. Nathan VanHoudnos

January 2020

Introduction

We thank the authors at the National Institute of Standards and Technology (NIST) National Cybersecurity Center of Excellence (NCCoE) for the opportunity to comment on Draft NISTIR 8269. We hope the authors find these comments useful, and invite them to contact us for any questions or clarifications.

NISTIR 8269 summarizes the state of the art of the academic literature in adversarial machine learning, focusing on supervised machine learning and classification problems, using the terms in the academic literature. Because of the important role that this taxonomy will play in shaping policy and research, we encourage NIST to adopt a broader perspective. Our specific recommendations are:

1. The academic adversarial machine learning literature focuses on the properties of different algorithms, rather than the diverse goals adversaries might have in attacking a system. We encourage adopting the language of security policies to explicitly tie adversarial actions to system vulnerabilities. This perspective presents a better framework for securing systems against a variety of adversarial threats.

2. A deployed ML system has a broader attack surface than is considered from a research perspective. We encourage NIST to re-examine the taxonomy from an operational perspective to identify gaps and omissions.

3. The academic literature reviewed for the taxonomy focuses on classification algorithms which are only a small fraction of machine learning algorithms. We encourage NIST to broaden the taxonomy to highlight the gaps in the academic literature. This will encourage clarity of thought and drive future research.

4. The language and ideas used in cybersecurity communities (such as government, academia, and various industry verticals) differ from those used in academic adversarial machine learning. These communities will need to work closely together in order to build trustworthy ML systems. We encourage NIST to take this opportunity to build the framework for this collaboration.

Risks in AML

The cybersecurity community has come to an agreement that there is no such question as to whether a system or model is secure in a general sense, only that a system or model can be secure relative to a particular security policy.[1] Consider the following examples where the exact same ML model is deployed in different contexts. The ML model counts the number of persons present in a given frame of video, and it has a logging feature that saves an image thumbnail every time a person is counted. This logging feature may be turned on or off.

Access Use-Case. The ML component is used to provide additional security for a badge-and-pin door. The security policy states that a person may enter the door only if they badge in and enter their PIN. The logging feature helps audit of the system.

Store Use-Case. The ML component is used to count the number of customers waiting in line at a particular national pharmacy chain. This information is used to predict staffing needs and for business marketing. The security policy states that only authorized persons should be able to access the count, and that the identities of the individuals in the line must be protected. The security policy additionally requires that the logging feature be turned off, to protect the identities of individuals and avoid privacy violations.

Table 1: A comparison of two ML use-cases.

		Security Policy	**Accuracy of ML component**
Access Control Use-Case	Failure to detect a person	Results in unauthorized access	Only matters insofar as security policy is violated
	Configuration of logging feature	No impact	Logging to monitor accuracy
Store Use-Case	Failure to detect a person	No impact	Affects business utility of the measurement
	Configuration of logging feature	No logging to protect privacy	No impact

In these two scenarios, the primary goals of an adversary will differ; moreover, what constitutes an adversarial attack and a security violation also differ. In the Access Use-Case, the goal of an adversary could be to gain unauthorized access to the facility. In the Store Use-Case, any person has access to the

[1] See, for example, the NIST definition of security (https://csrc.nist.gov/glossary/term/security) or the IETF definition of information security (https://tools.ietf.org/html/rfc4949). In general, if two systems have different functions, then we should expect an adversary attacking system A to have a different goal than system B. Thus, the two systems will need to be secure against different sets of attacks, to the extent where a necessary feature in one setting may be a security violation in another setting. A security policy makes these needs explicit. This is also why we should never label an ML system "secure," but rather make statements of the form "this system is secure in an environment where an attacker can attempt attack X against system resource Y."

facility; instead, the goal of an adversary could be to turn the logging feature on and gain access to sensitive information. The security policies take this into account and specify which parts of the system must be protected. Note that this is an example where a feature that helps secure the system in one context causes a security violation in the other, and why a system is never "secure" but only secure against specific attacks.

Now consider a specific weakness in the ML model. Suppose that winter hats impact the ability of the system to detect persons. In the Access Use-Case, a person wearing a winter hat is a security risk because an adversary could wear a hat in order to tailgate a person with legitimate access. It is not, however, a security risk in the Store Use-Case because the security policy of the store is focused on protecting the information gathered by the system, not assuring performance levels. For the Store Use-Case, the winter hat is an issue of robustness. We might worry about bias; that an error in the ML system would make it difficult to compare stores in warm regions to stores located in regions with colder climates for customer service purposes, but the error in the ML system in this scenario is not a security violation.

The intentions of the person wearing the hat are not what makes an outcome a security violation. If, for example, an adversary used recently published methods to make an adversarial hat (https://arxiv.org/abs/1908.08705) to fool this specific vision system, then for the Access Use-Case, the adversarially crafted hat creates a security violation, but in the Store Use-Case, even the adversarially crafted hat is only a robustness concern.

Operational Perspective

Draft NISTIR 8269 presents an academic/research perspective on ML systems as evidenced by the list of primary sources cited (lines 241-244). We suggest that NIST broaden the scope of the taxonomy to include an operational perspective. This operational perspective will also help when understanding the full risks of an ML system, and its relationship to any particular security policy.

One way to discuss an operational perspective is to give a broader representation of the machine learning process. We find Figure 1 to give a helpful representation of an operational machine learning system, since it can represent both supervised and unsupervised learning systems succinctly. Please note that for reinforcement learning, the general pipeline is the same, but the model building and validation stage will require modification.

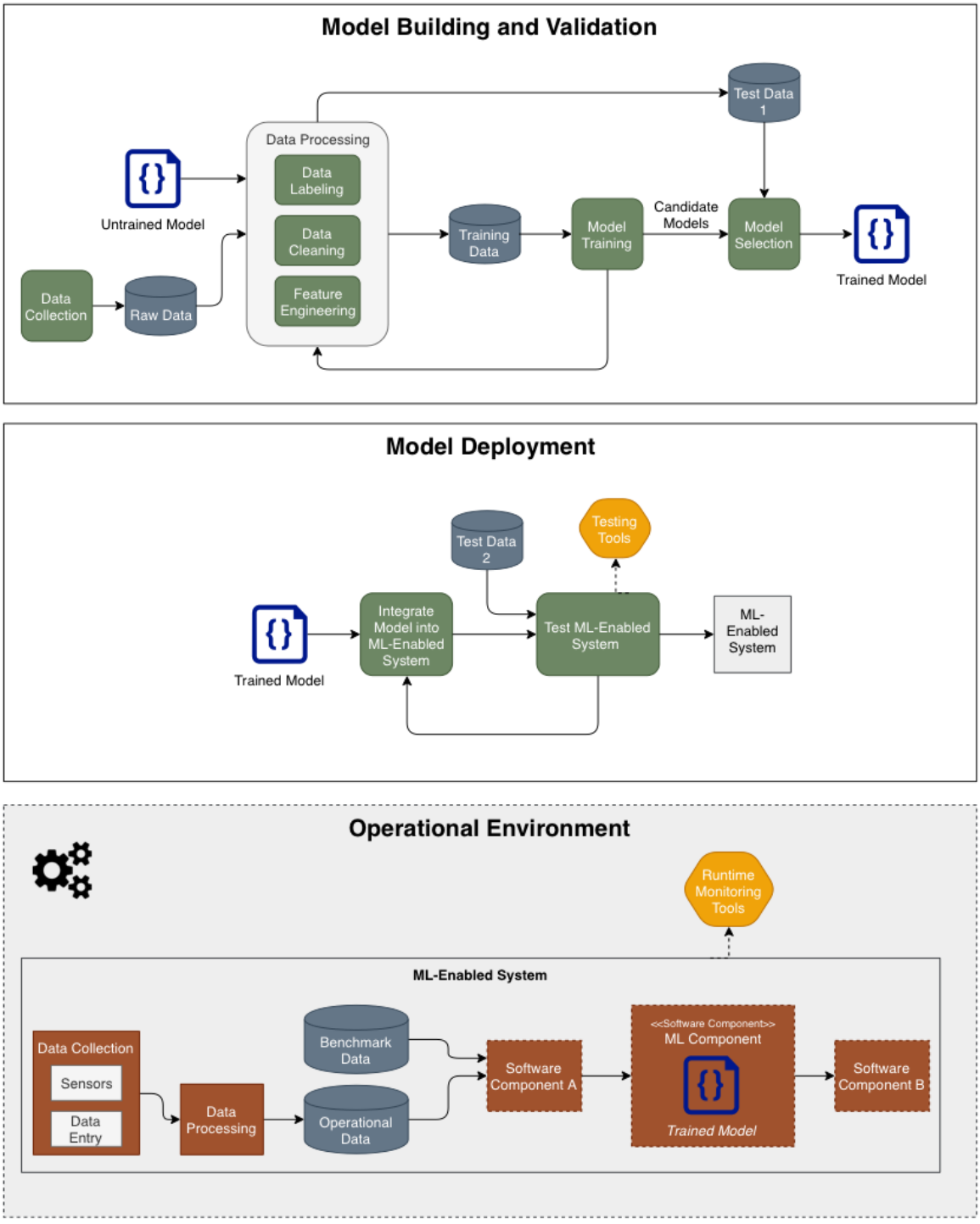

Figure 1: Each of the green boxes represents a process, often with a human directly involved. Each of the burnt-orange boxes represents a software component in the operational environment. In an operational ML system, the model needs frequent updating, and this process could be represented by adding a loop to this diagram.

One difference between the ML system representations in Draft NISTIR 8269 and Figure 1 is the inclusion of data sets beyond training and test data. These additional data considerations allow for a greater understanding of the attack space of your machine learning system. Specifically, Figure 1 shows six different datasets that are used in different ways: Raw Data is the data as it is collected from sensors. Model building includes an iterative process of cleaning, filtering, labeling, and feature engineering that produces the eventual Training Data and Test Data 1. The Training Data is the data used to train the model. Test Data 1 is a validation set that is used to verify that the model is functional and can generalize beyond the training data. During model deployment, the ML model is embedded in a software component that can be integrated into the full system. During this stage the developers will use Test Data 2 to validate the software implementation of the model. In many cases, Test Data 2 has a great deal of overlap with the Training Data or Test Data 1. Finally, in the Operational Environment, there is the Operational Data, for which the deployed model should be generating some insight, and the Benchmark Data, which is used to validate that the system is functioning as intended.

Concretely, if an adversary wanted you to waste time and money retraining your model, thus hurting you in a resource-constrained environment, they could attack the benchmark data to make your model seem as if performance was degrading for unknown reasons. On the converse, an adversary could attack the benchmark data to make the model seem as if it were performing well, when the truth was that performance had degraded; this kind of attack could cause high confidence in untrustworthy intelligence. These are both very different attacks than attacking the operational data directly, and could present different risks under different security policies.

In addition to the different types of data, the system representation of Figure 1 also highlights additional attacks that are not covered in the taxonomy. For example, in the two attacks on the benchmark data described above, the first attack is a *confidence reduction attack* in the draft taxonomy, but there is *no label* for the second attack. We believe that a thorough examination from this operational perspective will identify similar omissions. Each of the components and processes in the diagram represents a different avenue that an adversary could attack. They could attack the sensors that collect the data, the data processing component, or the runtime monitoring tools, in addition to attacking the model itself.

This operational systems perspective also highlights avenues for making the whole system more robust. If an adversary tinkers directly with a sensor, but adequate feature engineering is in place, then the adversary may not be able to achieve their goals and objectives by such tinkering. This example reflects a broader concept of systems security or security engineering.[2] A system is more than the sum of its parts, both in that someone may (1) build a secure system from vulnerable parts or (2) build an insecure system from secure parts. System security is a challenge because there are no guaranteed

---

2 See, for example, NIST SP 800-160 or the canonical textbook by Ross Anderson, *Security Engineering* (2008).

ways to do (1) or avoid (2). Security properties are not formally composable. Composability is an active area of research within cybersecurity.[3] It would be valuable for NISTIR 8269 to roadmap how AML should integrate with and support operational systems security more generally.

Over-Emphasis on Misclassification

The AML literature reviewed in draft NISTIR 8269 focuses on supervised learning, but more specifically classification problems (lines 360-361). There is good reason for this: while attacks on unsupervised learning systems and reinforcement learning systems have been studied,[4,5] the majority of the AML literature does indeed focus on supervised learning. The nascent state of the field has led many academic researchers to focus on investigating how different classification models respond to different specific permutations of the data. Such investigation is necessary and important research for making algorithms more robust, but it does not reflect the full attack surface and the cybersecurity landscape that we would expect to see in a deployed ML system, as discussed in the previous sections.

Our concern is that the taxonomy does not fully adjust for this imbalance in the research literature. For example, the section on attacks in the inference phase (lines 383-386) describes attacks that are applicable only to supervised learning systems, but this limitation in the definitions is *implicit*. It may better serve the interests of NIST's stakeholders to explicitly acknowledge that the taxonomy has a gap here because attacks in the inference phase on unsupervised learning systems and reinforcement learning systems are not as well understood. Leaving this gap unacknowledged may lead stakeholders to underestimate their risk for non-supervised learning because the definitions in the taxonomy do not prompt them to consider these cases outside of supervised learning.

The following examples illustrate the pervasiveness of this gap, but are not exhaustive:

- Ensemble learning or method is defined as *"A classification method using multiple classifiers to enhance robustness including against evasion attacks."* This definition is inappropriately narrow. A more common definition of ensemble method is *a technique that combines the predictions from multiple machine learning algorithms*. Indeed, one of the most common ensemble methods, random forests, can be used equally well with regression or classification.
- *Indiscriminate Attack* is defined as *"An attack that aims to cause misclassification of any sample to target any system user or protected service."* But we must ask whether an attack of this nature can truly only be applied to a classification system. It should be possible to attack a regression

[3] See, for example, the National Academies of Sciences, Engineering, and Medicine report "Foundational Cybersecurity Research" (2017).

[4] "Adversarial Policies: Attacking Deep Reinforcement Learning" https://arxiv.org/abs/1905.10615

[5] Chen, T., Liu, J., Xiang, Y. *et al.* Adversarial attack and defense in reinforcement learning-from AI security view. *Cybersecur* **2,** 11 (2019) doi:10.1186/s42400-019-0027-x

system so that any input receives an inaccurate output. It should also be possible to attack a clustering (unsupervised) system so that inputs are grouped inappropriately. This definition must be broadened.

- The definition of *Adversarial Example (ML input sample formed by applying a small but intentionally worst-case perturbation to a clean example, such that the perturbed input causes a learned model to output an incorrect answer.)* is inappropriately narrow. Must the perturbation be worst-case? By what metric are you measuring "worst"? Can I only input adversarial examples at inference time? Could adversarial examples not be included as part of a training set? Or is an adversarial example a particular observation which is designed to cause a ML system to output an incorrect inference?
    - In addition, this definition implicitly applies only to misclassification problems. For example, line 441: "In *Data Sanitization*, adversarial examples are identified by testing the impacts of examples on classification performance." This implies that the only place to find adversarial examples is in a classification system. (Note also that data sanitization should not be restricted to classification systems.)
    - This implicit limitation of "Adversarial Example" to misclassification can also be observed in related definitions: *Adversarial example transferability* is defined as *"The property that adversarial examples crafted to be misclassified by a model are likely to be misclassified by a different model."* If an adversarial example can exist for any type of ML system, then the idea of transferability between models must also exist for other types of ML systems.
    - This definition is particularly problematic because of downstream effects. For example, *Attack Detector* is defined as "a mechanism to detect if a piece of data is an adversarial example," so it now implicitly also applies only to a particular mode of attacking a classification model.

- Definition of "Poisoning Attack*" (Aims to increase the number of misclassified samples at test time by injecting a small fraction of carefully designed adversarial samples into the training data.)* is specific to a particular injection of a particular kind of sample for a misclassification goal. One could also inject malicious samples into training data for an unsupervised problem, but the current definition of poisoning attack does not give that kind of injection a place in the taxonomy.

The literature dealing with attacks on classification ML systems is more mature. This means that when we are specifically talking about attacks against a classification system, we can be precise in a way that we are not yet able to be with other systems. For example, definitions 30-33 (*Error-generic evasion attack* through *Error-specific poisoning attack*) come from the AML literature on misclassification, and these are very precise and useful distinctions. The fact that we do not yet have the same precision of language for attacks on other types of ML systems (e.g., clustering systems), reveals a gap in the literature.

We recommend that NISTIR 8269 strive for clarity about which definitions apply to which kinds of ML systems. This will prevent ambiguity and confusion; moreover, the presence of gaps in the literature can be used to drive research in AML.

Finally, it is worth noting that because supervised, unsupervised, and reinforcement learning algorithms address fundamentally different problems, they will be deployed in different contexts. For example, a supervised learning system might be deployed at a checkpoint to identify individuals who are allowed to pass, while an unsupervised learning system might be deployed to prioritize incoming intelligence for human analysts to review. Thus, the objectives of an adversary attacking these systems will likely be quite different. The taxonomy needs to have the flexibility to accommodate these differences in the full threat landscape. Incorporating the language of cybersecurity, particularly in relation to a security policy will help address this need.

Standard Cybersecurity Terms

The preceding discussion of risks in a machine learning system highlights the utility of cybersecurity-related terms in an AML setting. Ideas such as a security policy, a security incident, an adversary's objectives, and tactics, techniques, and procedures (TTP) will all play a role in any operational ML system. However, the adversarial machine learning (AML) community uses cybersecurity-related terms significantly differently than other cybersecurity communities.

While there is no single, universally accepted glossary for cybersecurity (or information security, information and communications technology security, etc.) different communities have collected their own standard or recommended glossaries. These include the IETF (RFC 4949), ISO (27000 series), IEEE (24765-2017), FIRST (e.g., CSIRT services framework), and the United States military (e.g., Joint Publication 3-12). NISTIR 8269 should not have to integrate the AML community with each possible, and possibly conflicting, definition for cybersecurity terms across each standard. However, we hope the NISTIR will choose to recommend how the AML literature might better communicate with cybersecurity communities by adopting a shared terminology.

Given that NIST is responsible for maintaining the cybersecurity standards for the U.S. federal civilian government it is well positioned to recommend synchronous definitions. As written, NISTIR 8269 proposes definitions that reflect the academic AML literature, and consequently conflict with established and accepted NIST definitions of terms from cybersecurity. We suggest resolving this state of conflict by explicitly mapping the academic AML definitions to the NIST cybersecurity definitions within the document.

For example, the NIST Computer Security Research Center maintains a searchable glossary that indexes all cybersecurity terms across all NIST publications (https://csrc.nist.gov/glossary). Table 2 provides examples of terms that we encourage NIST to deconflict with cybersecurity usage; we do not claim this list is exhaustive. Although we recognize that adopting this suggestion may require re-evaluation of portions of the AML taxonomy, we believe that synchrony of terms between AML and cybersecurity would benefit both research and practice.

Any term in Table 2 that is listed twice (as the NISTIR 8269 term and the accepted NIST term) means that NISTIR 8269 is using the term significantly differently than NIST's recommended cybersecurity definitions, with one exception (*threat*).

In Table 2, we recommend *adversary goals and objectives* as a missing term. We draw this definition from the Diamond Model,[6] which – within some cybersecurity communities – is the de facto standard on how campaign analysis is done. Campaign analysis has a primary goal of understanding adversary goals and objectives. The diamond model builds on the kill chain, a de facto standard for modeling individual attacks (in the second sense of attack).[7] We are not aware of an equivalent NIST term. We searched in the NIST CSRC glossary via expert heuristics, and did not find an equivalent term. The following terms were considered, but we think they are inadequate to capture "adversary goals and objectives:" Action-on-objectives; Adversary goal/objective; Attacker goal/objective; Goal; Objective; Outcome.

Table 2: Cybersecurity terms with conflicted usage in NISTIR 8269

NISTIR 8269 term	Accepted NIST term	Source document	Recommended Definition (in quotes) or comments
[missing]	Security policy	CNSSI 4009 OR SP 800-192	"A set of criteria for the provision of security services." OR "The statement of required protection for the information objects."
[missing]	Security service	SP 800-95	"A processing or communication service that is provided by a system to give a specific kind of protection to resources, where said resources may reside with said system or reside with other systems, for example, an authentication service or a PKI-based document attribution and authentication service. A security service is a superset of AAA services. Security services typically implement portions of security policies and are implemented via security mechanisms." Note, there are many security services besides the CIA triad. Authorization and non-repudiation are common additions, though some sources (e.g., RFC 4949) define more than 10 security services.
[missing]	Attack (second sense)	RFC 4949	"An intentional act by which an entity attempts to evade security services and violate the security policy of a system. That is, an actual assault on system security that derives from an intelligent threat." In NISTIR 8269, "attack" is used in a narrow sense to mean method an adversary uses. That sense is better captured by TTP. However, the second sense quoted here may need to be carefully introduced in relation to when an adversary in fact takes an action against a system.

6 The diamond model is introduced and defined in: Sergio Caltagirone, Andrew Pendergast, Christopher Betz. The Diamond Model of Intrusion Analysis. DTIC. 2014.

7 The claim these are de facto standards is supported in: Spring and Illari. Review of human decision-making during computer security incident response. arXiv. 2018.

NISTIR 8269 term	Accepted NIST term	Source document	Recommended Definition (in quotes) or comments
[missing]	Adversary goals and objectives	Diamond model (pg 20)	"[T]he social-political needs and aspirations of the adversary (e.g., to generate income, to gain acceptance in the hacker community, to become a hegemon, to increase business profits). The [adversary-victim] relationship denotes the need(s) of the adversary and the ability of the victim to satisfy the need(s) defining adversary intent (e.g., economic espionage, traditional espionage, criminal fraud, denial of service attack, website defacement). The victim unwittingly provides a "product" (e.g., computing resources & bandwidth as a zombie in a botnet, a target for publicity, industrial or business sensitive information for economic espionage, financial information and username/passwords for fraud) while the adversary "consumes" their product."
Adversary	Adversary	SP 800-107r1	An entity that is not authorized to access or modify information, or who works to defeat any protections afforded the information.
Attack	Tactics, techniques, procedures (TTPs)	SP 800-150	"The behavior of an actor. A tactic is the highest-level description of this behavior, while techniques give a more detailed description of behavior in the context of a tactic, and procedures an even lower-level, highly detailed description in the context of a technique."
Attacker	Adversary	SP 800-107r1	While "attacker" is defined in NIST SP 800-63-3, the definition requires knowing the actor's intentions. The adversary definition is based on authorization and protections, which are defined by the defender and so knowable by the defender. Working from knowable, if not known, measures should be preferred.
Availability	Availability, see also security service	44 U.S.C., Sec 3542	"Ensuring timely and reliable access to and use of information."
Confidentiality	Confidentiality, see also security service	44 U.S.C., Sec 3542	"Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information."
Consequence	Security incident	FIPS 200	"An occurrence that actually or potentially jeopardizes the confidentiality, integrity, or availability of an information system or the information the system processes, stores, or transmits or that constitutes a violation or imminent threat of violation of security policies, security procedures, or acceptable use policies."
Defense	Countermeasure; Security control	CNSSI 4009	"The protective measures prescribed to meet the security requirements (i.e., confidentiality, integrity, and availability) specified for an information system. Safeguards may include security features, management constraints, personnel security, and security of physical structures, areas, and devices." (security requirements come from the security policy)
Insider or outsider	See adversary	SP 800-53r4	Insider and outsider are types of adversaries. The key part of the definition is that an inside adversary is "within the security domain" (see security policy) or has "authorized access" (see security service). Any adversary not an insider is an outside adversary.
Integrity	Data Integrity, see also security service	44 U.S.C., Sec 3542	"Guarding against improper information modification or destruction, and includes ensuring information non-repudiation and authenticity."

NISTIR 8269 term	Accepted NIST term	Source document	Recommended Definition (in quotes) or comments
Poisoning	[none]		This term may conflict with specific types of network attacks, such as DNS cache poisoning, unless "poisoning" is further specified. Also, as noted above, the definition of "poisoning" in NISTIR 8269 is specific to an attack on a supervised learning system, but it is equally possible to commit a data poisoning attack on an unsupervised learning system.
Privacy	[unclear]	Privacy defined in SP 800-32 See also legal literature	"Restricting access to subscriber or Relying Party information in accordance with Federal law and Agency policy." As this definition implies, privacy is to a large extent synchronizing organizational policy or laws with security policy and therefore provision of adequate security services such as confidentiality. However, privacy cannot be reduced to NIST definitions. The Universal Declaration of Human Rights defines privacy as a human right (article 12). Various jurisdictions across the world interpret and implement this legal concept differently. Even the SP 800-32 definition makes the NISTIR 8269 reduction of privacy to a type of confidentiality violation (line 509) questionable, but the broader legal usage of the term suggests the authors may want to use a different term.
Security (sense 1)	Cybersecurity	CNSSI 4009	"Prevention of damage to, protection of, and restoration of computers, electronic communications systems, electronic communications services, wire communication, and electronic communication, including information contained therein, to ensure its availability, integrity, authentication, confidentiality, and nonrepudiation."
Security (sense 2)	Security	CNSSI 4009	"A condition that results from the establishment and maintenance of protective measures that enable an enterprise to perform its mission or critical functions despite risks posed by threats to its use of information systems. Protective measures may involve a combination of deterrence, avoidance, prevention, detection, recovery, and correction that should form part of the enterprise's risk management approach."
Target	Information system	44 U.S.C., Sec 3502	"A discrete set of information resources organized for the collection, processing, maintenance, use, sharing, dissemination, or disposition of information." Note that "target," though italicized on line 46, is not defined in section 3.
Technique	See TTPs	N/A	MITRE's ATT&CK framework and the NISTIR use this term differently: MITRE's usage is similar to that in TTP, while the NISTIR uses this term to mean an attack or attack type.
Threat	Threat	CNSSI 4009	"Any circumstance or event with the potential to adversely impact organizational operations (including mission, functions, image, or reputation), organizational assets, individuals, other organizations, or the Nation through an information system via unauthorized access, destruction, disclosure, modification of information, and/or denial of service." The most important feature of the threat definition to highlight is that threats are any circumstance with potential. Attacks (sense 2, not in the sense of TTPs) are actual adversary actions, whereas threats are potential adversary actions. The NISTIR captures the potential nature of "threat" but should incorporate the CNSSI definition.
Vulnerability	Vulnerability	CNSSI 4009 (adapted)	"a weakness in an information system, including in its system security procedures, internal controls, requirements, design, or implementation, that could be exploited or triggered by a threat source."

Contact Us

Software Engineering Institute
4500 Fifth Avenue, Pittsburgh, PA 15213-2612

Phone: 412/268.5800 | 888.201.4479
Web: www.sei.cmu.edu
Email: info@sei.cmu.edu

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

This report was prepared for the SEI Administrative Agent AFLCMC/AZS 5 Eglin Street Hanscom AFB, MA 01731-2100

DM20-0070