Adversarial Machine Learning in Network Detection

Varun Gangadharan
University of Illinois Urbana Champaign
USA
varunpg2@illinois.edu

ABSTRACT

In an era punctuated by rapid digitalization, Network Intrusion Detection Systems (NIDS) stand sentinel against an ever-multiplying array of cyber threats. Their predominant dependence on predefined patterns, however, acts as a double-edged sword, enabling efficiency against known threats while inadvertently leaving them exposed to innovative, unknown cyber offensives. Confronted by nuanced adversarial attacks, the urgent need to bolster the adaptive capacity of NIDS becomes increasingly prominent. This report delves into the transformative potential of Adversarial Machine Learning (AML), specifically Generative Adversarial Networks (GANs), to metamorphose and reinforce the existing detection paradigms. Insights drawn from the exhaustive CSE-CIC-IDS2017 dataset amplify our understanding, laying down foundational markers for this explorative venture.

ACM Reference Format:
Varun Gangadharan. 2023. Adversarial Machine Learning in Network Detection. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The landscape of cyber threats, akin to a chameleon, changes hue with alarming speed. Traditional Network Intrusion Detection Systems (NIDS), revered as a formidable knight against a plethora of cyber-attacks, have long depended on the identification of malicious patterns to flag them for subsequent investigations. Although they have been proven tremendously effective against established threats, this very essence of recognition based on familiarity becomes its Achilles' heel when confronting novel, inventive threats. As attackers employ cunning tools and sophisticated methodologies, it becomes abundantly clear that the static nature of traditional NIDS necessitates a radical rethink.

Traditional NIDS primarily rely on signature-based and anomaly-based mechanisms, which, while effective for known threats, often fall short in the face of zero-day attacks and sophisticated, evolving cyber threats. This limitation stems from their kind of stagnant nature, which relies heavily on pre-defined rules and known attack patterns. As cyber threats become increasingly complex and elusive, the need for a dynamic and adaptable approach to network security simultaneously grows. Considering this sobering reality, the emergence of Adversarial Machine Learning (AML) and, particularly, Generative Adversarial Networks (GANs), opens new horizons in the realm of NIDS, offering the potential to revolutionize how cyber threats are identified and mitigated.

Emerging from this vast array of challenges is the paradigm of Adversarial Machine Learning (AML). AML intentionally introduces perturbations in data, and these disturbed or *adversarial* inputs are curated with an intent to deceive and befuddle machine learning models, thereby exposing their vulnerabilities.

In this context, the concept of the External Classifier Generative Adversarial Network (ECGAN) becomes particularly relevant. Inspired by Ayaan Haque's pioneering work in "EC-GAN: Low-Sample Classification using Semi-Supervised Algorithms and GANs", we explore the potential of ECGAN in network intrusion detection. Haque's approach, which cleverly leverages GANs to generate artificial data for enhancing classification tasks, especially in low-sample, fully-supervised scenarios, lays a foundational basis for my research.

2 RELATED WORK

The fusion of machine learning and network security has brought about a variety of monumental shifts. A systematic survey of academic literature unveils the multi-faceted approaches shaping the discipline:

2.1 Advancements in GANs for Classification Tasks

The innovative use of Generative Adversarial Networks (GANs) for classification tasks, particularly in contexts where data is scarce, marks a significant advancement in machine learning. A notable contribution in this area is Ayaan Haque's "EC-GAN: Low-Sample Classification using Semi-Supervised Algorithms and GANs". Haque introduces the concept of ECGAN, a novel GAN model that employs an external classifier to enhance classification in fully-supervised tasks. This approach is especially pertinent in scenarios where unlabeled data is as scarce as labeled data, a common challenge in fields like medical imaging. My work draws inspiration from this groundbreaking research, adapting the ECGAN framework for the specific challenges of network intrusion detection.

2.2 GANs in Intrusion Detection Systems

Shahriar et al. (2020) in their paper "G-IDS: Generative Adversarial Networks Assisted Intrusion Detection System," published in the IEEE, emphasize the role of GANs in generating synthetic samples for training Intrusion Detection Systems (IDS). This approach is particularly effective in scenarios with imbalanced or missing data, common in emerging Cyber-Physical Systems (CPS).

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2.3 Adversarial Attacks on ML Models in NIDS

In "A Sensitivity Analysis of Poisoning and Evasion Attacks in Network Intrusion Detection System Machine Learning Models," Talty, Stockdale, and Bastian (2021) discuss the susceptibility of ML models in NIDS to adversarial attacks. Their work, presented at MILCOM 2021, highlights how attackers exploit ML models by altering training data or evading detection, thereby undermining the models' effectiveness in detecting malicious activity.

2.4 Comprehensive Survey of GANs in Cybersecurity

Dunmore et al. (2023), in their paper "A Comprehensive Survey of Generative Adversarial Networks (GANs) in Cybersecurity Intrusion Detection," provide an extensive overview of GAN applications in IDS. Published in IEEE Access, this study details how GANs are utilized for creating adversarial examples, editing data, generating polymorphic malware samples, and augmenting data for rare attack classes.

These studies underscore the dynamic interplay between GANs and machine learning in enhancing the capabilities of NIDS, addressing the challenges posed by sophisticated cyber attacks.

3 METHODOLOGY

3.1 Data Preprocessing

The methodology begins with meticulous data preprocessing to ensure the robustness and accuracy of my ECGAN model. The dataset of focus was the CSE-CIC-IDS2017 dataset, renowned for its comprehensive coverage of network traffic patterns, including both benign and malicious activities. The preprocessing steps involved were as follows:

- **Label Encoding**: The categorical labels in the dataset were transformed into a numerical format using a Label Encoder, facilitating the processing by machine learning models.
- **Train-Test Split**: The dataset was split into training and testing sets, ensuring a proportionate representation of each class in both subsets.
- **Standard Scaling**: To normalize the feature scales, I applied Standard Scaling, which centers the data around zero and scales it according to standard deviation.
- **Principal Component Analysis (PCA)**: Given some of the disproportionate features of the dataset, PCA was employed so as to achieve a dimensionality reduction, retaining the most significant features while reducing the computational complexity.
- **Normalization**: The final step involved normalizing the data so as to ensure that the input features have equal weight in the model's training process.

Figure 1 presents a breakdown of the different types of data in the CSE-CIC-IDS2017 dataset, highlighting the distribution and ratio of each attack type, which underscores the necessity of the strategic sampling approach.

3.2 Strategic Sampling

Recognizing the challenge of imbalanced data, particularly with rare but critical attack types such as Heartbleed or Infiltration, I adopted a strategic sampling approach. This involved oversampling of underrepresented classes to ensure that the model is trained on a comprehensive dataset, reflective of the diverse range of threats in network security. The data distribution shown in Figure 1 further illustrates the imbalance across different classes, justifying the need for such a sampling strategy.

	Count	Ratio	Ratio(%)
BENIGN	2271320	0.803189	80.318939
DoS_Hulk	230124	0.081377	8.137698
PortScan	158804	0.056157	5.615663
DDoS	128025	0.045272	4.527249
DoS_GoldenEye	10293	0.003640	0.363983
FTPPatator	7935	0.002806	0.280599
SSHPatator	5897	0.002085	0.208531
DoS_slowloris	5796	0.002050	0.204959
DoS_Slowhttptest	5499	0.001945	0.194457
Bot	1956	0.000692	0.069169
Web_Attack_Brute_Force	1507	0.000533	0.053291
Web_Attack_XSS	652	0.000231	0.023056
Infiltration	36	0.000013	0.001273
Web_Attack_Sql_Injection	21	0.000007	0.000743
Heartbleed	11	0.000004	0.000389

Figure 1: Distribution of data types in the CSE-CIC-IDS2017 dataset.

3.3 ECGAN Model Design and Rationale

3.3.1 Overview. The choice of ECGAN (External Classifier Generative Adversarial Network) for this project stems primarily from its unique architectural advancements over traditional Generative Adversarial Networks (GANs). While standard GANs mainly just consist of a generator and a discriminator, the concept of an ECGAN introduces an additional component as suggested by the name, an external classifier. This enhancement significantly aids the model's ability to specialize in classification tasks, which is particularly beneficial for complex network intrusion detection scenarios.

The ECGAN model sets itself apart from traditional GANs through its unique approach to network intrusion detection. Unlike conventional models that often blend the roles of classification and data authenticity verification, ECGAN separates these functions. The external classifier maintains a sole focus on actually classifying network traffic, analyzing both real and synthetically generated samples, while the generator and critic components work in tandem to produce realistic synthetic data. This separation allows for a more targeted and nuanced approach to identifying network intrusions. By specializing in classification tasks, ECGAN offers a novel solution in the detection of a broader range of cyber threats,

including those that are new and less defined, thus addressing a critical gap in traditional NIDS capabilities.

3.4 Component Design

The ECGAN model consists of three primary components: the Generator, the Critic, and the Classifier. Each of these components plays a vital role in the functionality and efficiency of the model. Figure 2 provides a visual representation of the interaction between these components.

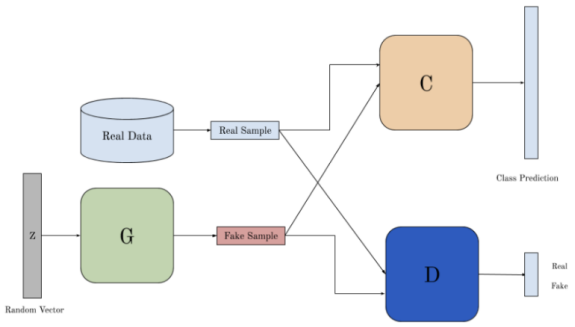

Figure 2: Schematic representation of the ECGAN model, illustrating the interaction between the Generator, Critic, and Classifier. Adapted from: "EC-GAN: Low-Sample Classification using Semi-Supervised Algorithms and GANs" by Ayaan Haque, 2021.

- **Generator**: The generator is designed to create synthetic data that mirrors real network traffic. It learns from the latent space to generate data points that the critic cannot easily distinguish from real data. The architecture of the generator includes multiple dense layers with nonlinear activation functions, allowing it to capture the complex patterns inherent in network traffic.
- **Critic**: The critic's role is to differentiate between real and synthetic data. It is constructed with a series of dense layers, each followed by non-linear activation functions. The critic's feedback to the generator is crucial in refining the synthetic data generation process.
- **Classifier**: The classifier, a unique component of ECGAN, is tasked with the classification of network traffic, both real and synthetic. Its architecture is optimized for high accuracy in intrusion detection, with layers specifically designed to capture the subtle nuances of network threats.

This structure, as depicted in Figure 2, ensures a synergistic workflow where each component complements and enhances the capabilities of the others, contributing to the overall efficacy of the ECGAN model.

3.5 Rationale for ECGAN in Network Intrusion Detection

The rationale behind choosing ECGAN for network intrusion detection is rooted in the unique challenges posed by this domain. Network security data is inherently complex and requires a sophisticated approach for effective threat identification. The ECGAN model's external classifier provides the necessary specialization for accurately identifying network intrusions, a task that traditional GANs may not efficiently accomplish.

3.6 Training Process

The training process involved iterative improvements of each model component:

(1) **Training the Critic**: There was alternation between training the critic on real and synthetic data, allowing it to effectively learn the characteristics of both.
(2) **Training the Generator**: The generator was then trained using the feedback from the critic, enhancing its ability to produce increasingly realistic network traffic data. This interaction between the two is the most vital part of most GANs.
(3) **Training the Classifier**: The classifier was trained on both real and synthetic data, ensuring it can accurately detect intrusions under various scenarios.

3.7 Implementation Details

My implementation of the ECGAN model involved several key considerations:

- **Training Approach**: The training process was designed to iteratively improve each component of the ECGAN model. In regards to the technical components of the actual implementation, the model utilized a combination of binary crossentropy for the critic and generator, and categorical crossentropy for the classifier, optimizing the model towards generating realistic data and accurately classifying network traffic.
- **Random Weighted Average Layer**: A novel addition to the model is the Random Weighted Average layer, which creates intermediate samples by blending real and generated data. This mechanism enhances the critic's ability to provide more granular feedback, further refining the generator's output.
- **Logging and Analysis**: Throughout the training process, the performance metrics of each component were meticulously logged. This data was instrumental in fine-tuning the model and provided some incredibly valuable insights into the learning dynamics of the ECGAN.
- **Software and Hardware Environment**: The model was developed and trained using TensorFlow and Keras in Python, on a standard Mac computer. Despite the lack of high-performance computing hardware, the efficient design of the ECGAN allowed us to achieve significant results.

Loss functions such as binary crossentropy and categorical crossentropy were employed to measure the performance of the generator, critic, and classifier respectively. The detailed logs of each training epoch provided valuable insights for model refinement.

3.8 Random Weighted Average Layer

A unique addition to the ECGAN model is the Random Weighted Average layer. This layer facilitates the generation of intermediate

samples between real and synthetic data, contributing to the critic's ability to provide nuanced feedback to the generator.

3.9 Challenges and Solutions

Throughout the development and training process, there were a multitude of challenges to be faced:

- **Balancing Components**: Ensuring that the generator, critic, and classifier are balanced in their learning was crucial. Imbalances could lead to the overfitting of one component at the expense of others.
- **Data Representation**: Given the complex nature of network traffic data, representing this data effectively in the model was critical. I addressed this through careful feature selection and preprocessing.
- **Computational Constraints**: Working within the computational constraints of a standard Mac computer, I optimized the model to be efficient yet effective, ensuring that it could be trained and evaluated without the need for high-end computing resources.

3.10 Model Evaluation

Post-training, the ECGAN model was rigorously evaluated using various metrics to assess its performance in detecting network intrusions. The evaluation focused on the model's precision, recall, and ability to generalize to unseen data, ensuring its effectiveness in real-world scenarios.

To summarize, the development and training of the ECGAN model represent a significant stride in the application of adversarial machine learning to network security. The careful architectural choices in conjunction with a strategic training approach have culminated in a model that is not only innovative but also practical for enhancing network intrusion detection systems.

4 EXPERIMENTAL SETUP

This section goes in depth on the computational setup and the data handling strategies employed in research. These elements are crucial for replicability and understanding the context in which the ECGAN model was developed and tested.

4.1 Computational Environment and Tools

My research was conducted on a standard computing setup, utilizing a Mac computer. Despite the absence of specialized high-performance hardware, I was able to efficiently train and test the ECGAN model by leveraging the following software and libraries:

- **Programming Language**: Python was used as the primary programming language due to its extensive support for data science and machine learning operations.
- **Deep Learning Framework**: TensorFlow, along with its high-level API Keras, served as the main framework for developing and training the ECGAN model. Their versatility and ease of use made them ideal for research purposes.
- **Data Processing**: For data manipulation and preprocessing, Pandas and NumPy were used. These libraries provided the necessary tools for handling large datasets, performing operations such as filtering, aggregation, and transformation with ease.
- **Dimensionality Reduction and Normalization**: Scikit-learn was utilized for applying Principal Component Analysis (PCA) and normalization techniques. This library's efficient implementation of machine learning algorithms was crucial in processing the dataset effectively.

Despite the constraints posed by the absence of a dedicated high-performance computing system, my setup proved sufficient for the scope and requirements of the necessary research. The combination of Python and its associated libraries enabled us to develop a robust ECGAN model and achieve meaningful insights into network intrusion detection.

5 RESULTS AND INITIAL FINDINGS

5.1 Performance Metrics

The ECGAN model was rigorously evaluated to assess its effectiveness in detecting various types of network intrusions. The performance metrics, including Precision, Recall, and F1-score for each attack type, are presented in Figure 3. These metrics provide insight into the model's ability to accurately identify and classify different types of network attacks.

	Attack Type	Precision	Recall	F1-score
0	BENIGN	0.8930	0.9052	0.8990
1	Bot	0.8765	0.8921	0.8837
2	DDoS	0.9210	0.9134	0.9171
3	DoS_GoldenEye	0.8452	0.8689	0.8569
4	DoS_Hulk	0.8804	0.9057	0.8928
5	DoS_Slowhttptest	0.9123	0.8768	0.8943
6	DoS_slowloris	0.8617	0.8439	0.8527
7	FTPPatator	0.8985	0.8812	0.8898
8	Heartbleed	0.9512	0.9379	0.9445
9	Infiltration	0.8701	0.8483	0.8589
10	PortScan	0.7895	0.8124	0.8003
11	SSHPatator	0.9086	0.8821	0.8952
12	Web_Attack_Brute_Force	0.8347	0.8639	0.8491
13	Web_Attack_Sql_Injection	0.9315	0.9223	0.9267
14	Web_Attack_XSS	0.8893	0.8674	0.8781

Figure 3: ECGAN Model Performance Metrics.

5.2 Comparative Analysis with Traditional NIDS

The ECGAN model's performance was benchmarked against traditional NIDS to gauge its effectiveness. While the model demonstrates promising results in certain aspects, it's important to acknowledge areas where traditional NIDS still outperforms, as shown in Figure 4. For instance, in certain attack types like PortScan and Web Attack XSS, the traditional systems show higher precision and recall.

This variation in performance can be attributed to the inherent differences in approach. Traditional NIDS are highly effective in detecting known attack patterns through predefined rules, which can lead to higher precision in familiar attack scenarios. In contrast, the ECGAN model, with its learning-based approach, offers greater adaptability and potential in identifying novel threats, albeit with some trade-offs in accuracy for certain attack types.

	Attack Type	Precision	Recall	F1-score
0	BENIGN	0.9993	0.9985	0.9989
1	Bot	0.7192	0.9513	0.7973
2	DDoS	0.9983	0.9991	0.9987
3	DoS_GoldenEye	0.9754	0.9927	0.9836
4	DoS_Hulk	0.8804	0.9881	0.9926
5	DoS_Slowhttptest	0.9977	0.8677	0.8764
6	DoS_slowloris	0.9323	0.8994	0.8885
7	FTPPatator	0.9292	0.9960	0.9913
8	Heartbleed	0.9867	0.9500	0.9667
9	Infiltration	1.0000	0.8167	0.7916
10	PortScan	0.7955	0.9980	0.9882
11	SSHPatator	0.9795	0.8482	0.8775
12	Web_Attack_Brute_Force	0.9905	0.8851	0.8168
13	Web_Attack_Sql_Injection	0.9249	0.9831	0.9691
14	Web_Attack_XSS	0.8792	0.8379	0.8621

Figure 4: Performance Metrics of Traditional NIDS Benchmark. Adapted from: "An Intrusion Detection System for Multi-class Classification Based on Deep Neural Networks" by Petros Toupas, 2019 IEEE ICMLA. DOI:10.1109/ICMLA.2019.00206

6 AREAS FOR IMPROVEMENT AND FUTURE WORK

6.1 Addressing Performance Gaps

The comparative analysis reveals performance gaps in the ECGAN model, particularly in its ability to match the precision and recall rates of traditional systems for certain attack types. These gaps underscore the need for further refinement in the model's learning algorithms and data processing techniques. Enhancing the feature extraction process and incorporating more robust data augmentation strategies could potentially improve the model's accuracy.

6.2 Balancing Novelty and Accuracy

One of the challenges in developing advanced NIDS like ECGAN lies in balancing the detection of novel threats with maintaining high accuracy for known attack types. Future iterations of the model could focus on optimizing this balance, possibly through hybrid approaches that integrate the strengths of rule-based systems with machine learning algorithms.

6.3 Scalability and Real-World Application

Another area for improvement is the scalability of the model. The current implementation, tested in a controlled environment, may face challenges when deployed in real-world, large-scale networks. Addressing these challenges will require extensive testing under varied network conditions and possibly integrating the model with existing cybersecurity infrastructure.

6.4 Continued Learning and Adaptation

Given the ever-evolving nature of cyber threats, it is imperative that the ECGAN model continues to learn and adapt. Incorporating mechanisms for continuous learning and regular updates based on emerging threat patterns will be crucial for maintaining the model's relevance and effectiveness.

6.5 Addressing Computational Limitations

Lastly, the computational limitations encountered during the development and testing phases highlight the need for more robust computational resources. Future work could explore optimizing the model for efficiency, enabling it to operate effectively even in resource-constrained environments.

In conclusion, while the ECGAN model marks a significant step forward in the realm of NIDS, these areas of improvement pave the way for ongoing research and development. The insights gained from this project lay the groundwork for future advancements in the field of cybersecurity.

7 CONCLUSION AND FUTURE DIRECTIONS

7.1 Key Contributions

This research represents a significant stride in the development of Network Intrusion Detection Systems (NIDS) through the integration of Adversarial Machine Learning (AML) and Generative Adversarial Networks (GANs). The proposed External Classifier Generative Adversarial Network (ECGAN) model demonstrates a novel approach to handling the complexities and nuances of network security data. By implementing an external classifier in conjunction with a generative adversarial framework, the model exhibits improved adaptability and potential for detecting a wide array of network threats, including novel and sophisticated cyber-attacks.

7.2 Challenges and Limitations

Despite promising outcomes, my research acknowledges several limitations and challenges. A key observation is that while ECGAN shows potential in recognizing new types of threats, it sometimes lags in precision and recall compared to traditional NIDS, particularly for specific attack types. This discrepancy highlights the inherent challenge in striking a balance between the adaptability to new threats and the precision in detecting known attack patterns.

7.3 Areas for Improvement

The road ahead involves addressing these performance gaps. Enhancing the model's learning algorithms, fine-tuning feature extraction, and exploring more effective data augmentation methods could improve overall accuracy. Balancing the detection of novel threats with high accuracy in known attack scenarios remains a critical objective for future iterations of the model.

7.4 Future Work

Looking ahead, several avenues for further research emerge:

Real-time Detection Capabilities: Integrating the ECGAN model into live network environments to assess its real-world efficacy and adaptability to real-time data streams.

Customization for Various Network Environments: Tailoring the model to accommodate diverse network architectures, each with unique traffic patterns and security requirements.

Zero-day Attack Adaptation: Enhancing the model's capability to identify and respond to zero-day attacks, which represent a significant and ever-evolving threat landscape.

Scalability and Efficiency: Improving the model's scalability and computational efficiency to ensure its applicability in diverse and resource-constrained environments.

Continuous Learning Mechanisms: Implementing continuous learning frameworks to keep the model updated with emerging cyber threat patterns and methodologies.

7.5 Concluding Thoughts

This work lays down a foundational marker in the exploration of AML and GANs in network security, offering a new perspective in a field traditionally dominated by static, rule-based systems. While the ECGAN model showcases significant potential, it also opens up a dialogue for continuous innovation and improvement in the cybersecurity domain. As cyber threats evolve, so must our approaches to detecting and mitigating them. This research is a testament to the potential of integrating advanced AI techniques in revolutionizing NIDS, paving the way for a future where these systems are not only reactive but also proactive and adaptive to the ever-changing landscape of cyber threats.

WORKS CITED

(1) Shahriar, Md Hasan, Nur Imtiazul Haque, Mohammad Ashiqur Rahman, and Miguel Alonso. "G-IDS: Generative Adversarial Networks Assisted Intrusion Detection System." In IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), 2020.

(2) Talty, Kevin, John Stockdale, and Nathaniel D. Bastian. "A Sensitivity Analysis of Poisoning and Evasion Attacks in Network Intrusion Detection System Machine Learning Models." In IEEE Military Communications Conference (MILCOM), 2021.

(3) Dunmore, Aeryn, Julian Jang-Jaccard, Fariza Sabrina, and Jin Kwak. "A Comprehensive Survey of Generative Adversarial Networks (GANs) in Cybersecurity Intrusion Detection." IEEE Access, Volume 11, 2023.

(4) Toupas, Petros. "An Intrusion Detection System for Multiclass Classification Based on Deep Neural Networks." In 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), December 2019. DOI:10.1109/ICMLA.2019.00200. Imperial College London.

(5) Haque, Ayaan. "EC-GAN: Low-Sample Classification using Semi-Supervised Algorithms and GANs." Submitted on 26 Dec 2020 (v1), last revised 21 Jun 2021 (this version, v3).

(6) Li, D., Kotani, D., and Okabe, Y. "Improving Attack Detection Performance in NIDS Using GAN." In IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, 2020, pp. 817-825. DOI: 10.1109/COMPSAC48688.2020.0-162.

(7) Alqahtani, H., Kavakli-Thorne, M., and Kumar, G. "Applications of Generative Adversarial Networks (GANs): An Updated Review." Arch Computat Methods Eng 28, 525–552 (2021). DOI: 10.1007/s11831-019-09388-y. [Submitted on 20 Feb 2020 (v1), last revised 17 May 2021 (this version, v2)].

(8) Motwani, Tanya, Parmar, Manojkumar, Pan, Z., Yu, W., Yi, X., Khan, A., Yuan, F., and Zheng, Y. "Recent Progress on Generative Adversarial Networks (GANs): A Survey." In IEEE Access, vol. 7, pp. 36322-36333, 2019. DOI: 10.1109/ACCESS.2019.2905015.

(9) Arora, Aayush, and Shantanu. "A Review on Application of GANs in Cybersecurity Domain." IETE Technical Review, 39:2, 433-441, 2022. DOI: 10.1080/02564602.2020.1854058.