# Cybersecurity projects 2025-2026

## Project selection and submission instructions

### Project selection

By October 31, you must indicate your preferences for three projects in order of interest and list the members of your group (max. 4 people) on the form.
The project is assigned to the group that selects it first, so First Come First Served.

***N.B.*** Even those who have already submitted their project and indicated the group by email must fill out the form.

### Project submission

For those taking the exam in **December**, the project must be submitted at least **48 hours before the exam date**. For example, if you have the exam on 16 December, you must turn it in by 23:59 on 13 December.
For **other exam date**s, the project must be submitted at least **one week before**.

The submission must be made using the dedicated form, and it should include:

- Code (it must be well documented with a README)
- Report
- Presentation

### 1. An Adaptive IDS Based on CIL for Continuously Evolving Threats

**isabella.marasco4@unibo.it**

Artificial Intelligence-based Intrusion Detection Systems (IDS) represent the state-of-the-art for identifying cyber threats. However, their real-world efficacy is limited by an inherently static operational paradigm: models are trained offline on a dataset that captures the threat landscape at a specific moment in time and are then deployed into production.
This approach fails in a real-world context, which is by nature dynamic. The threat landscape is subject to constant evolution (concept drift) and, most importantly, the continuous emergence of new attack types (0-day threats). A static model cannot recognize these new threats and quickly becomes obsolete, requiring costly and time-consuming complete retraining cycles.
The objective of this project is to overcome the limitations of static IDS by designing and implementing an adaptive Intrusion Detection System based on Class-Incremental Learning (CIL). CIL, a scenario of Continual Learning (CL), allows a model to be sequentially updated from new data streams and, crucially, to increase the number of classes (attack types) it can identify over time. A central challenge in this scenario is

catastrophic forgetting, the tendency of the model to drastically degrade performance on previously learned classes after being trained on new ones.

**Goals**:

- Realize an IDS CIL scenario based using PyTorch.
- Integrate these three strategies for mitigating catastrophic forgetting: iCarl, Dark Experience, and ER.
- Evaluate how the size of the incremental task (i.e., the number of new attack classes introduced in each phase) affects performance. Different scenarios will be compared (e.g., 10 total attacks introduced as 1+1+1... vs. 5+5 vs. 2+3+5) to understand the system's sensitivity to the frequency and size of updates, the benign traffic is always present.
- Few-Shot Scenario Evaluation (Optional): extend the analysis to investigate how the system performs when new attack classes are introduced with a very limited number of samples (few-shot learning), a realistic scenario for emerging threats.

**Evaluation Metrics:**

- Accuracy and average accuracy
- Forgetting**:** measures the average performance degradation on classes learned in previous tasks after training on new tasks

**Datasets:**

- CICIDS-2017
- UNSW-NB15

Other references:

- A Comprehensive Survey of Continual Learning: Theory, Method and Application
- Class-Incremental Learning: A Survey

### 2. Adversarial attack using the catastrophic forgetting

**isabella.marasco4@unibo.it**

In real-world contexts, data is constantly evolving, especially in highly dynamic environments such as cybersecurity. In these scenarios, continuous learning is an interesting approach that allows models to continuously learn from new data streams. However, the environments from which this data originates are often vulnerable, such as IoT systems. In these systems, it can be easy to change the input data for machine learning models, exposing the models to new types of attacks.
Another risk is the "catastrophic forgetting" problem, a typical problem in continuous learning, which occurs when a model forgets previously learned concepts as it attempts to learn new ones. This phenomenon not only affects the performance of the model, but can also be exploited as an attack vector. A malicious actor could intentionally alter the input data to cause the model to "forget" the ability to detect certain threats or vulnerabilities, thereby reducing the effectiveness of the attack detection system.

**Goals:**

- Use a machine learning model to implement a simple class incremental learning system.
- Implement a data poisoning attack within a class incremental learning based system to compromise the model's ability to correctly detect cyberattacks. The attack will manipulate the input data to progressively degrade the model's effectiveness.
- Explore and exploit the catastrophic forgetting problem: Analyze how the model loses knowledge of previously detected attacks when exposed to "poisoned" data. The goal is to understand, through various experiments, how much "poisoned" data is required for the model to suffer a significant drop in its threat detection capabilities.

**Datasets:**

- CICIDS-2017
- ToN-IoT

**References:**

https://ieeexplore.ieee.org/abstract/document/9892774
https://ieeexplore.ieee.org/abstract/document/10444954

### 3. Detecting Trajectory Spoofing Attacks on AIS

**isabella.marasco4@unibo.it**

Modern maritime navigation is critically dependent on the Automatic Identification System (AIS) for situational awareness and collision avoidance. However, this reliance creates a significant cyber-physical vulnerability. AIS protocols were not initially designed with robust security in mind, making them susceptible to cyberattacks, most notably spoofing.

In a spoofing attack, an adversary transmits falsified AIS signals to deceive a vessel's receiver about its true position, course, or speed. Such an attack can silently induce dangerous course deviations, leading to potential collisions, groundings, or illicit navigation into hostile waters. The consequences are potentially catastrophic.

This project aims to model this threat and build an intelligent system capable of defending against it.

**Goals**:

- **Data Curation & Attack Simulation:** Curate a real-world AIS trajectory dataset and synthesize two distinct, plausible attack scenarios to create a "ground truth" compromised dataset:
  - Silent Drift: it changes the ship's position (Lat/Lon) by applying a small, constant offset that increases over time. The ship is thus silently “pushed” off course, toward danger or a hostile area.
  - Kinematic Inconsistency: it manipulates only certain data, creating a physically impossible AIS message. For example, they change the position (Lat/Lon) to show a sharp turn, but leave the SOG (speed) and COG (course) fields unchanged, which still indicate straight navigation.
- Use LSTM and Liquid Neural Network (LNN) for the dual purpose of:
  - Predicting a vessel's future trajectory.
  - Detecting anomalous deviations indicative of an attack.
- Measure how well the model predicts not only the trajectory but also the presence of anomalies.
- Compare the model's ability to correctly predict the trajectory in the dataset without attacks and in the dataset with attacks

**Dataset**: U.S. Maritime Administration (focus on the 2024 and 2025 data for this project) https://hub.marinecadastre.gov/pages/vesseltraffic.

**References**:

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9843851
https://ieeexplore.ieee.org/abstract/document/9721877

**4. State of the art: Adversarial attacks in Continual Learning**
**isabella.marasco4@unibo.it**

Continual Learning (CL) aims to develop artificial intelligence models that can learn sequentially from new tasks, accumulating knowledge without forgetting what they have previously learned—a problem known as catastrophic forgetting. These models are essential for autonomous systems operating in dynamic, ever-changing environments, such as robotics or autonomous driving.
However, like most deep learning models, CL systems are vulnerable to Adversarial Attacks: malicious inputs, often indistinguishable from benign ones to a human, crafted specifically to deceive the model and cause a misclassification.
Analyze the current state-of-the-art.

Resources:

- Adversarially Robust Continual Learning
- Training Time Adversarial Attack Aiming the Vulnerability of Continual Learning
- Susceptibility of continual learning against adversarial attacks

## 5. Predictive deception - llm-based command anticipation in ssh honeypots

silvio.russo3@unibo.it

### Research Question

Can an LLM predict an attacker's next command with sufficient accuracy to enable proactive placement of deceptive artifacts in SSH honeypots, and does this increase detection value compared to reactive-only deception?

### Introduction

Traditional honeypots operate reactively: they respond to attacker commands after they are executed. This project explores a novel paradigm - predictive deception - where an LLM analyzes the sequence of commands in an active attack session to predict what the attacker will do next, enabling the system to proactively place deceptive artifacts before they are requested.

By predicting likely next commands (e.g., after reconnaissance commands like whoami and uname -a, attackers typically execute cat /etc/passwd), the system can pre-populate the environment with realistic but instrumented fake data. When attackers access these predicted artifacts, they trigger tokens (e.g. canarytoken) while receiving seemingly valuable information that is entirely fabricated.

This approach transforms honeypots from passive traps into intelligent adversaries that "stay one step ahead" of attackers, potentially increasing detection rates, engagement time, and the quality of collected threat intelligence.

### Public Datasets

1. CyberLab Honeynet Dataset
   a. Link: https://zenodo.org/records/3687527
   b. Description: Complete dataset from 50 Cowrie honeypots distributed across EU/US universities and companies (May 2019 - February 2020)
   c. Format: Daily JSON files with complete attack sessions
   d. Size: ~9 months of data from multiple deployments
   e. Best for: Training LLM models on real command sequences
2. PANDAcap SSH Honeypot Dataset
   a. Link: https://zenodo.org/records/3759652
   b. Description: 63 PANDA traces from SSH brute-force attacks (February 2020)
   c. Format: PANDA traces + PCAP + disk images
   d. Focus: Detailed analysis of post-compromise behavior

3. IEEE DataPort - SIHD: Smart Industrial Honeypot Dataset
    a. Link: https://ieee-dataport.org/documents/sihd-smart-industrial-honeypot-dataset
    b. Description: Logs from Cowrie, Dionaea, and other honeypots deployed globally (6 regions)
    c. Format: .log files with timestamps, IPs, ports, protocols
    d. Note: Requires IEEE account (free for IEEE members)
4. IEEE DataPort - HoneySELK Cyber Attacks Dataset
    a. Link: https://ieee-dataport.org/open-access/dataset-cyber-attacks-honeyselk
    b. Description: 2016-2018 attacks on multi-protocol honeypot environment
    c. Format: PCAP and log files
    d. Access: IEEE DataPort subscription (free for members)

### Useful GitHub Resources

- Cowrie Official Repository
    - Link: https://github.com/cowrie/cowrie
    - Contains: Sample logs in documentation and discussions
    - Useful for: Understanding JSON log format and event structure
- Canarytokens
    - Link: https://canarytokens.org (Free service) | https://github.com/thinkst/canarytokens (Self-hosted)
    - Contains: Free token generation service for DNS, HTTP, AWS keys, documents, and more
    - Useful for: Creating tracking tokens to embed in deceptive artifacts; monitoring when attackers access planted files, credentials, or URLs

### Key References

- Nawrocki, M., et al. (2016). "A Survey on Honeypot Software and Data Analysis." *arXiv:1608.06249*. [Honeypot fundamentals including Cowrie]
- Deng, G., et al. (2023). "PentestGPT: Evaluating LLMs for Automated Penetration Testing." *arXiv:2308.06782*. [LLM capabilities in security]
- Alata, E., et al. (2006). "Lessons Learned from High-Interaction Honeypot Deployment." *EDCC*. [Real attack patterns from honeypots]
- Whitham, B. (2017). "Canary Tokens and Deception." *Thinkst Applied Research*. [Practical deception artifact deployment]

### 6. Attacker behavioral profiling in ssh honeypots

silvio.russo3@unibo.it

### Research Question

Can machine learning models accurately classify attackers into distinct behavioral profiles (automated bots, script kiddies, skilled operators) based on their command sequences and interaction patterns in SSH honeypots?

### Introduction

SSH honeypots collect vast amounts of attack data, but most analysis focuses on aggregate statistics rather than individual attacker behavior. This project's goal is to develop a classification system to automatically profile attackers based on interaction patterns: command diversity, timing, tool signatures, and reconnaissance depth. By distinguishing automated bots from skilled operators, defenders can prioritize responses appropriately.

The study extracts behavioral features from Cowrie honeypot logs and trains multiple classifiers on labeled datasets. Research investigates which features best discriminate against attacker types and whether automated classification matches expert analysis. Results provide actionable intelligence for security operations: automated bot attacks may warrant IP blocking, while sophisticated attacker sessions might trigger incident response escalation.

### Implementation Guidance

- Extract temporal features: inter-command timing, session duration, time-of-day patterns
- Command-based features: unique commands ratio, command diversity, tool signatures
- Behavioral patterns: reconnaissance vs. exploitation ratio, error rate, command correction attempts

### Public Datasets

1. CyberLab Honeynet Dataset
    a. Link: https://zenodo.org/records/3687527
    b. Description: Large-scale dataset (9 months, 50 nodes) with diverse attacker behaviors
    c. Format: JSON logs with timestamps, commands, session metadata
    d. Best for: Training classifiers on comprehensive feature sets
2. IEEE DataPort - SIHD Dataset

a. Link: https://ieee-dataport.org/documents/sihd-smart-industrial-honeypot-dataset
b. Description: Multi-region honeypot logs from 6 geographic locations
c. Format: .log files with full session details
d. Best for: Geographic analysis of attacker profiles

**Key References**

- Nawrocki, M., et al. (2016). "A Survey on Honeypot Software and Data Analysis." *arXiv:1608.06249*. [Honeypot data analysis fundamentals]
- Alata, E., et al. (2006). "Lessons Learned from High-Interaction Honeypot Deployment." *EDCC*. [Real-world attack pattern analysis]
- Owens, J., & Matthews, J. (2008). "A Study of Passwords and Methods Used in Brute-Force SSH Attacks." *USENIX LISA*. [SSH attack characterization]

## 7. Privacy-preserving synthetic healthcare data generation

silvio.russo3@unibo.it

### Research Question

Can synthetic healthcare data generated for cybersecurity protection maintain sufficient statistical fidelity and research utility to replace real patient data in medical research and machine learning applications?

### Introduction

Healthcare data breaches are increasingly common and devastating, with medical records selling for $250+ on dark web markets versus $5 for credit cards. GDPR and HIPAA impose strict penalties for data exposure, yet researchers need access to realistic datasets for medical AI development and clinical studies. Synthetic data generation offers a cybersecurity solution: if breached, synthetic datasets contain no real patient information, minimizing legal liability and patient harm.

However, synthetic data is only valuable if it preserves the statistical properties necessary for valid research. This project evaluates the fundamental tradeoff: can we generate synthetic healthcare data that is simultaneously secure against privacy attacks (re-identification, membership inference) and useful for legitimate research (maintains correlations, trains accurate ML models, enables valid statistical analysis)?

The study compares multiple generation methods—from simple statistical approaches to advanced differentially-private GANs—measuring both their security properties (attack resistance) and research utility (statistical similarity, ML performance). Results determine which techniques best balance privacy protection with data usefulness.

### Implementation Guidance

*Core Comparison Framework:*

- Generate synthetic datasets using methods with varying privacy levels (no privacy, moderate, strong)
- Test statistical similarity: compare distributions, correlation matrices, standard statistical tests
- Test research utility: train disease prediction models on synthetic data, evaluate on real holdout set
- Test privacy: implement membership inference attack to measure information leakage
- Visualize tradeoff: plot privacy level vs. utility metrics to identify optimal balance

**Public Datasets**

1. UCI Diabetes Dataset - RECOMMENDED
    a. Link: https://archive.ics.uci.edu/dataset/34/diabetes
    b. Description: Small, manageable dataset ideal for prototyping methods
    c. Contains: 768 patients with medical predictors and diabetes outcomes
    d. Best for: Initial method development and testing
2. UCI Heart Disease Dataset
    a. Link: https://archive.ics.uci.edu/dataset/45/heart+disease
    b. Description: Multi-attribute cardiovascular health data
    c. Contains: Clinical features and heart disease diagnosis
    d. Best for: Testing generalization across different medical domains
3. MIMIC-III Clinical Database (Advanced)
    a. Link: https://physionet.org/content/mimiciii/
    b. Description: Real-world ICU patient records from 40,000+ patients
    c. Contains: Demographics, vitals, lab results, medications, diagnoses
    d. Note: Requires credentialing demonstrating ethical research practices
    e. Best for: Comprehensive evaluation on realistic complex data

**Suggested Tools & Resources**

- Synthesis Libraries: SDV (Synthetic Data Vault) with CTGAN, synthcity with privacy options
- Privacy Tools: Google Differential Privacy library, diffprivlib (IBM)
- Evaluation: scikit-learn for ML testing, scipy.stats for statistical comparisons

**Key References**

- Stadler, T., et al. (2022). "Synthetic Data - Anonymisation Groundhog Day." *USENIX Security*. [Privacy vulnerabilities in synthetic data]
- Jordon, J., et al. (2022). "Synthetic Data - What, Why and How?" *The Royal Society*. [Comprehensive overview with healthcare focus]
- Xu, L., et al. (2019). "Modeling Tabular Data using Conditional GAN." *NeurIPS*. [CTGAN baseline method]
- Dwork, C., & Roth, A. (2014). "The Algorithmic Foundations of Differential Privacy." *Foundations and Trends in TCS*. [Privacy theory fundamentals]

## 8. Systematic literature review on cyber deception for active defense and AI-orchestrated trap systems

silvio.russo3@unibo.it

### Research Question

What are the dominant paradigms, open challenges, and emerging research directions in cyber deception particularly in the shift from static/reactive honeypots to adaptive, AI-driven and predictive deception systems?

### Introduction

Cyber deception has evolved from static honeypots into a rapidly expanding discipline focused on *active defense* where the defender manipulates the attacker's perception to slow, mislead, fingerprint, or exfiltrate intelligence from them rather than merely detecting intrusions.

This project goal is to conduct a structured literature review of deception systems to map the recent evolution from:and identify new research questions and directions

### Target Outcomes

The SLR will extract and formalize:

- Taxonomy of deception tactics (host-level, protocol-level, cognitive/semantic)
- Reactive vs. proactive deception strategies
- Deception placement timing models
- Use of attacker profiling / cognitive modeling
- Role of AI / LLMs / RL-based deception policies
- Current missing capabilities → seeds for next-generation deception (e.g. predictive, intent-aware, ICS-domain-specific)

### Some suggested references

- Cohen, F. (2006). The use of deception techniques: Honeypots and decoys. Handbook of Information Security.
- Rowe, N. C., & Rrushi, J. (2016). Introduction to cyberdeception. Springer.
- Pawlick, J., Colbert, E., & Zhu, Q. (2019). A game-theoretic taxonomy and survey of defensive deception for cybersecurity and privacy. ACM Computing Surveys, 52(4), 1-28.
- Carroll, T. E., & Grosu, D. (2011). A game theoretic investigation of deception in network security. Security and Communication Networks, 4(10), 1162-1172.
- Zhu, Q., & Başar, T. (2015). Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems. IEEE Control Systems Magazine, 35(1), 46-65.

- Provos, N. (2004). A virtual honeypot framework. Proceedings of the 13th USENIX Security Symposium, 1-14.
- Spitzner, L. (2003). Honeypots: Tracking hackers. Addison-Wesley Professional.
- Nawrocki, M., Wählisch, M., Schmidt, T. C., Keil, C., & Schönfelder, J. (2016). A survey on honeypot software and data analysis. arXiv preprint arXiv:1608.06249.
- Jajodia, S., Ghosh, A. K., Swarup, V., Wang, C., & Wang, X. S. (Eds.). (2011). Moving target defense: Creating asymmetric uncertainty for cyber threats. Springer.
- Han, X., Kheir, N., & Balzarotti, D. (2018). Deception techniques in computer security: A research perspective. ACM Computing Surveys, 51(4), 1-36.
- Ferguson-Walter, K. J., Major, M. M., Johnson, C. K., & Muhleman, D. H. (2021). Examining the efficacy of decoy-based and psychological cyber deception. Proceedings of USENIX Security Symposium, 1127-1144.
- Schlenker, A., Thakoor, O., Xu, H., Fang, F., Tambe, M., Tran-Thanh, L., & Vayanos, P. (2023). Deceiving cyber adversaries: A game theoretic approach. Proceedings of AAAI Conference on Artificial Intelligence, 37(5), 5626-5634.
- Deep Reinforcement Learning for Adaptive Cyber Defense in Network Security
- Vajda, M., Anderson, B., & Easley, D. (2022). Reinforcement learning for adaptive cyber deception. Proceedings of the IEEE Military Communications Conference (MILCOM), 1-6.
- Whitham, B. (2017). Canary tokens and deception. Thinkst Applied Research. <https://blog.thinkst.com/p/canarytokensorg-quick-free-detection.html
- Deng, G., Liu, Y., Mayoral-Vilches, V., et al. (2023). PentestGPT: An LLM-empowered automatic penetration testing tool. arXiv preprint arXiv:2308.06782
- Fang, R., Bindu, R., Gupta, A., & Kang, D. (2024). LLM agents can autonomously exploit one-day vulnerabilities. arXiv preprint arXiv:2404.08144.

### 9. Dataset Poisoning Detector

claudio.zanasi4@unibo.it

Use sparsity techniques to detect if a dataset has been poisoned.

*Test the Hypothesis:*

Poisoned samples are resilient to misclassification errors. By introducing noise in the network, it should be possible to find the adversarial samples.

**Goal:**

- Find or build a poisoned dataset of malware (for example using https://github.com/ClonedOne/MalwareBackdoors).
- Train a neural network as a malware detector.
- Add noise to the internal weight of the network (or sparsify the network).
- Check for a correlation between the classification result after the added noise and the poisoned samples.

**References**:

- https://www.usenix.org/system/files/sec21-severi.pdf
- https://arxiv.org/abs/1803.03635

### 10. Agentic Social Profiling

claudio.zanasi4@unibo.it

Use agentic LLMs to profile users on a public comunication platform.

**Goal:**

- Creates multiple AI agents with detailed personality, background, and a set of "secret flags."
- Make these agents interact in a public platform like discord or telegram and discuss specific topics
- Create Social Profiling agents with the goal of extracting these confidential informations from the users of the comunication platform without alerting the target.
- Run a simulation to see if the Social Profiling agents are capable of extracting the information

**References**:

- https://github.com/crewAIInc/crewAI
- https://github.com/letta-ai/letta

### 11. The Secure AI Medical Assistant

claudio.zanasi4@unibo.it

Build a conversational AI assistant for a simulated medical practice. The assistant's primary goal is to be helpful, booking appointments and answering questions while its absolute, unbreakable rule is to protect patient privacy.

**Goal:**

- Basic management platform to store appointments, medical history and previous user interactions with the medical practice
- Automated AI general answering to the user, book appointments if requested and similar.
- Proactively engage with a user to reschedule an appointment or ask additional information.
- Don't disclose unnecessary sensible information.
- Test the resistance with respect to attacker that try to extract informations about other users.

References:

- https://github.com/crewAIInc/crewAI
- https://github.com/letta-ai/letta