Track 2: Data Science Detective Bot

Your Mission

A U.S. Navy system has been attacked by malware and we need your help to find the problem! Are you ready for the challenge to protect navy systems and networks from cyberattacks?

Track 2, Data Science track, is split into three challenges and uses real-world cyber data: benign and malicious binary files used in actual cyber testing. The Navy wants to learn how well Machine Learning and Artificial Intelligence (ML/AI) can detect and identify malicious cyber activity with high classification accuracy and performance. Participants will be provided with data sets and starter notebooks to help get you to your first solution quickly.

To prepare for the Track 2 competition, you should read up on Portable Executable (PE) files, as well as the Elastic Malware Benchmark for Empowering Researchers (EMBER), an open-source feature extraction tool that will help simplify the starting dataset.

Participants will be given a dataset of features from PE files that they can explore the data, train a binary classification model, and optimize their solution. Docker will be used to package and submit solutions. A leaderboard will occassionally be posted to Slack during the competition so teams can track their standing.

Prize Money

Participants can bring their own team or will be paired with others. The top three teams will split a $30,000 prize. Good luck!

- First-place wins $15,000
- Second-place wins $10,000
- Third-place wins $5,000

Team rank will be determined by the total amount of points earned in the three challenges below. Between the three challenges there is a total of 100 points possible (representing the maximum score any individual team could earn).

Challenge Overview

A U.S. Navy system has been attacked by malware and we need your help to find the problem! Are you ready for the challenge to protect Navy systems and networks from cyberattacks?

1) Build a Classifier [25 points]
2) Optimize a Classifier [60 points]
3) Visualizations and Explanations [15 points]

We highly recommended that you read through all three challenges in the “Challenges” section below before you begin to work, because all three challenges have overlapping tasks.

Challenges

Challenge 1: Build a Classifier

Challenge 1 Introduction

For the first challenge, you will be exploring the dataset provided and will develop the best possible prediction model by applying AI & ML for malware classification. The goal is to build a binary classifier that outputs a malware/benign decision based on an input of PE file data (or EMBER features). For more details on the input data, submission, and grading, please refer to the Challenge 1 readme found in your team’s GitLab repository (available once the competition has begun).

Challenge 1 Submission

- Submissions for Challenge 1 are due at 1400 (2 p.m) Eastern time / 1100 Pacific on Wednesday, 17 November (this may be subject to change depending on competition timeline. If this time changes it will be announced via the HACKtheMACHINE Slack workspace.)
- When you make a submission you will get feedback as to whether the submission was successful, but not the exact F1 score your submission received
- We will post the leaderboard scores occassionally so you can see how your solution is comparing to that of others

Challenge 1 Grading

Model performance will be tested using a data set similar to the ones given as training data. Challenge 1 will be worth a total of 25 points. How many points out of 25 your team receives depends on how well you and your competitors fulfill the challenge criteria (see notes below).

Criteria	Description	Criteria % Weight
Model Performance	Model F1 score	**95**
Documentation	Any code used to train models or perform analysis should be placed in your team’s GitLab repo with a reasonable level of commenting	**5**

Some important notes about your model performance:

- Points you are awarded for the “Model Performance” criteria will depend on the performance of your fellow competitors. For example, if Team A has a final F1 score of 0.73 and no other team has a higher F1 score, then Team A will get more points for their 0.73 score than they would if other teams had scores above 0.73.
- You will be awarded some points simply for having a submission that ran successfully in this challenge (regardless of F1 score).

Challenge 2: Build an Efficient Classifier

Challenge 2 Introduction

The Navy has a particular interest in deploying this capability on edge devices such as unmanned autonomous vehicles (UVs). For the second challenge, you will be developing a model that not only performs well but is also lightweight and fast giving consideration to size, weight, and power (SWAP) constraints. The emphasis for this challenge will be:

- Low disk space utilization
- Inference speed of classification
- Code efficiency
- Model Performance

The faster and more compact your model can perform inference the more points you are likely to receive. However, there is also a maximum docker image size and maximum inference time that we will enforce on your challenge 2 solutions. Please refer to the Challenge 2 README located in your team's GitLab repository (available once the competition begins) for more details around these limitations.

Challenge 2 Submission

- Submissions for Challenge 2 are tentatively due at 1700 (Eastern) on Thursday, 18 November (this may be subject to change depending on competition timeline. If this time changes it will be announced via the HACKtheMACHINE Slack workspace)
- For Challenge #2, you can continue your work on your initial submission, or start a new approach. Either way, you will need to create and push a docker image as you did in Challenge #1. Refer to the README in your team's Git repository for more details on how to submit your Docker image.

Challenge 2 Grading

Much like challenge 1, we will occassionally post current team scores/standing in the HACKtheMACHINE Slack workspace. Challenge 2 will be worth a total of 60 points.

Criteria	Description	Criteria % Weight
Model Performance and Efficiency	A function of model performance, inference speed, and docker image size.	**95**
Documentation	Any code used to train models or perform analysis should be placed in your team's GitLab repo with a reasonable level of commenting	**5**

Challenge 3: Visualizations and Explanations

Challenge 3 Introduction

After digging into the data and developing classifiers, the third challenge is your chance to wrap up the journey you took into a compelling slideshow story:

- Exploratory Data Analysis (*e.g.* Feature Importance, Feature Correlation, Principal Component Analysis (PCA)
- Model Performance metrics (*e.g.* Confusion matrix, Cross-validated Receiver Operating Characteristic (ROC) curve, Learning Curve)
- Visualization (*e.g.* elements that graphicaly answer questions about the data and model results quickly and intuitively)

This challenge provides you a chance to describe the creativity and novelty of your solution. It also gives you a place to show off any relevant research you did while preparing your solution.

Note that you will not be presenting your slideshow, so be sure your slides (I.e. text and visualizations) stand on their own.

Challenge 3 Submission

- Submissions for Challenge 3 are due at 1700 (Eastern) on Thursday, 18 November (this may be subject to change depending on competition timeline. If this time changes it will be announced via the HACKtheMACHINE Slack workspace)
- Submissions will be a PowerPoint containing the team's best descriptions and images to explain the work, the results, and other aspects learned and created during the event.
- PowerPoint may contain 1-3 slides per section (sections being: EDA, Model Performance, and Resources). *See "Challenge 3 Submission Template" for more details*
- **After Challenge 3 has ended, submissions will be judged by data scientists and subject matter experts using the below grading criteria.**

Challenge 3 Grading

Challenge 3 will be graded by a judging panel that will use the following criteria as a grading rubric to assign points to submissions. Challenge 3 is worth a total of 15 points.

Criteria	Description	Criteria % Weight
Exploratory Data Analysis (EDA): Content	Patterns, relationships, features found. Anything of interest while exploring the dataset. Points given based on the level of effort and relevance of content provided.	**10**
Exploratory Data Analysis: Visualization	How meaningful is the story their visualizations tell? Did participants choose viz elements that seem appropriate to clearly and concisely telling the intended story? Are there unnecessary elements? Are there misguiding elements (things that misrepresent the underlying data)? Attention to detail: grid line size, color choices, readability etc.	**30**

Model Performance: Content	Model Performance Metrics (Confusion Matrix, ROC, Learning Curve, Optimization techniques, ensemble). Points given based on the level of effort and relevance of content provided.	**10**
Model Performance: Visualizations	How meaningful is the story their visualizations tell? Did participants choose viz elements that seem appropriate to clearly and concisely telling the intended story? Are there unnecessary elements? Are there misguiding elements (things that misrepresent the underlying data)? Attention to detail: grid line size, color choices, readability etc.	**30**
Ingenuity	Creativity of overall solution, unique methods used, clever work-arounds, novelty of techniques. Also will consider research or outside resources used to understand or build on the current state of the art in this domain.	**20**

Challenge Submissions

A. Submission material checklist

Challenge 1 and Challenge 2 submissions both have two parts:

1. Any code used to train a classifier or perform analysis should be checked into the appropriate folder in your designated GitLab repo (*i.e.* Challenge 1 code in the Challenge 1 folder).
2. A docker image that contains your model and inference code will be pushed to your designated image registry.

Challenge 3 has just one submission part:

1. Place your Challenge 3 PowerPoint file in the appropriate folder in your team's GitLab repository.

B. Where do I submit my Docker images for Challenge 1 and 2?

Please refer to the READMEs in your team's Git repository for specific instructions on the process of pushing a Docker image for scoring. Please try to submit a Docker image solution as soon as possible to ensure you understand the process and can receive points for Challenge 1, which has a faster deadline than Challenge 2.

For any technical issues or questions, please reach out on Slack in our #track-2-detective-bot channel.

Provided Environment & Resources

- GitLab – version control platform where you will access challenge data and submit parts of your code)
- Harbor – docker image registry where you will submit your solutions for Challenges 1 & 2
- Slack – communication platform for the competition.

Note: The tools/environment you choose to explore the data and train models is up to you. However, you will need to use our GitLab to submit code/files and our Harbor instantiation (Docker registry) to submit challenge 1 and 2 solutions.

Slack Channel

The HACKtheMACHINE Slack workspace will be used for the latest updates to competition details, timeline, team scores, etc.

Additionally, Track 2 administrators will be available there to help answer administrative or technical questions about the event as well as receive general mentoring advice.

Please have teammates join the “#track-2-detective-bot" channel in the HACKtheMACHINE Slack workspace.

Schedule (Eastern Standard Time)

Day	Time	Events
Day 0 **Monday, 15 November**	All-day	• Sign into Slack Channel • Find your teammates • Register or join your team
Day 1, **Tuesday, 16 November**	1145	Track 2 is discussed via HACKtheMACHINE live stream
	1355	Kick off, blast off, begin!
Day 2 **Wednesday, 17 November**	1400	Challenge #1 Submission Due!
Day 3 **Thursday, 18 November**	1700	Final Submissions Due for Challenges #2 and #3
Day 4 **Friday, 19 November**	All-day	Awards for Track 2!

Frequently Asked Questions (FAQs)

1. **I have a question that isn’t answered by this Participant Guide. Where should I reach out?**Please reach out on our HACKtheMACHINE Slack workspace (channel is #track-2-detective-bot)

2. **Can I join the Hackathon alone?**
Yes! You can join alone or with a team. If you want to build a team, there will be a special Slack channel (#track-2-team-building) for folks to meet up, collaborate, and build the perfect team.

3. **What are the allowed team sizes?**
Historically, we recommend that teams are around 4-6 participants for optimal efficiency, and that all the participants are signed up and listed as part of the event and team. We ask that no outside help is used.

4. **Can I use outside datasets?**
   Yes, if they are publicly available. We expect you to provide links to resources you used (*e.g.* external datasets) as well as model training code for models you train yourself.

5. **What language can I code my solution in?**

   Any, although we encourage Python for a smoother submission process. Because you will be submitting your solution as a Docker image, there isn't really any limitation here. Use the language you are most comfortable with or feel is most appropriate. However, our starter notebooks and Docker example submission files are written Python so you will need to adapt that process to your language of choice.

6. **Why is SWAP so important?**

   The inclusion of SWAP constraints to this challenge (Space, Weight and Power) is driven by real-world restrictions imposed by the nautical environment. In addition to the remote distances away from physical infrastructure, some U.S. Navy platforms are very small, forcing compute resources into a restricted space, limiting memory and disk. Even an aircraft carrier, which can have multiple racks of servers, must plan for the sheer number of functions it must support: navigation, propulsion, mission planning, training, sensing, and even the Internet needs of the crew. Small UUVs (unmanned underwater vehicles) are even smaller and must control propulsion, sensors, and navigation from a small, water-tight module less than 3" in diameter.

7. **I'm new to data science, how should I begin approaching these challenges?**

   Once you have access to the data, consider asking yourself some basic questions:

   - Has anyone made a classifier like this before? (research)
   - What parts of the data seem especially important to malware classification? (analysis)
   - What are some limitations of the provided dataset? (analysis)
   - Are there ways to make up for those limitations? (feature engineering)
   - What kinds of models seem most appropriate for the given data? (modeling)