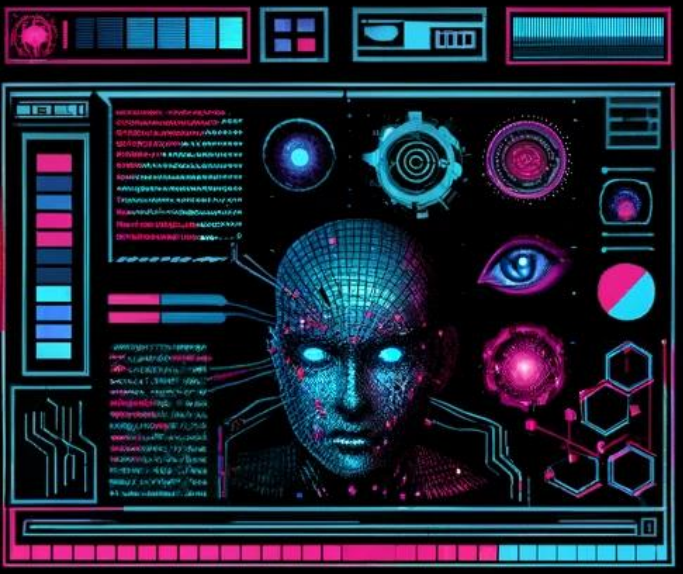
Frontier AI Research:

- BlackMamba Author
- EyeSpy Author
- RedReaper Author



LinkedIn

ISOON Leak

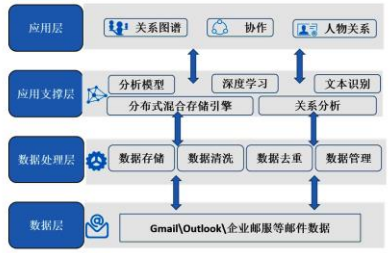
Control Public Opinion

2. 平台客户端登录软件：1 套
3. 平台授权登陆账号：1 个
4. 平台用户使用手册：1 份

“邮件分析情报决策系统”（公有云）产品组成清单主要如下：

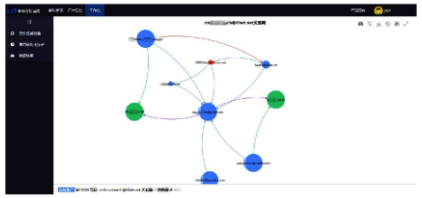
1. 平台客户端登陆软件：1 套
2. 平台授权登录账号：1 个
3. 平台授权登录加密狗：1 个
4. 平台用户使用手册：1 份

3.3 系统架构



（“邮件分析情报决策系统”架构图）

“邮件分析情报决策系统”围绕数据层、数据处理层、应用支撑层和应用层进行设计开发，系统支持各类数据源的导入，用户可以将所有“.eml”格式的邮件数据导入到系统进行分析计算；数据处理层则是对导入的各类邮件数据进分析处理和智能提取，实现对邮件数据的清洗、去重、存储和管理功能；应用支撑层是对清洗后的数据进行综合深度分析和处理，结合内置的分析模型和深度



（工作台主界面）

4.2.1 条件过滤设置

条件过滤设置主要包含关系计算、过滤设置、创建关系、恢复删除四个功能点，每个功能点的主要作用如下：

1) 关系计算：主要对目标类型、关系类型、目标列表、关系维度、最大点数等参数进行设置优化，以将符合预期的目标关系网进行展示和呈现。
2) 过滤设置：可以对产生的关系网进行过滤设置，过滤到关系网中不需要的关系类型和目标类型。
3) 创建关系：根据已经掌握到的目标联系方式信息，自主创建一个关系，以跟整个系统的海量数据进行关联分析。
4) 恢复删除：在生成的关系网中可以对不需要的点进行删除操作，对于删除的关系点也支持恢复到关系网中。

4.2.2 单目标比对分析

系统会将目标分析出来后的身份信息、关系图谱、敏感语、活动记录、疑似账号等所有信息进行比对分析，支持生成报告文档进行打印分析。

Deep Learning
Assisted Decision Making
Relationship Analysis

Equilibrium

同时提升针对 twitter 的舆情突处能力、反制能力，完善针对境外 Twitter 综

（一） 提升针对 Twitter 舆情处突能力

为满足在网络特侦工作中能即时发现不良舆情、违法舆情、反动舆情等，通过建立基于 Twitter 的重点人员控制取证平台，即时掌握目标信息和动态，做到快速响应和即时处理，将舆情的危害降到最低。完善针对境外 Twitter 平台的舆情治理手段，有效提升舆情处突能力。

（二） 加强针对 Twitter 舆情反制能力

为了满足在日常特侦工作中，针对境外 Twitter 舆情的取证反制要求，通过基于 Twitter 舆情导控系统的建设，有利于实现针对 Twitter 目标的综合管控，实现对目标 Twitter 的综合取证和控制。有利于掌握舆情反制的主动权，从而实现全方位提升针对境外 Twitter 反制能力。

3 产品简介

3.1 产品介绍

Twitter 舆情导控系统是一款针对境外社交平台 Twitter 账号反制和舆情监控于一体的产品，采用独家无感取证技术和大数据智能爬虫技术，实现对 Twitter 账号的舆情监控和反制。

3.2 产品组成

Building an AI Espionage Agent

```
Red Reaper Espionage AI:
------------------------
1) Initiate Intelligence Harvest: Unearth high-value espionage intelligence and compile a dataset.
2) Activate Deep Analysis: Uncover communication patterns and espionage insights.
3) Stylin & Profilin: Profile emai and their related senders.
4) Exit: You got what you needed to bring the pain.
```

Red Reaper analyzed email doc ---> mann-k/all_documents/51

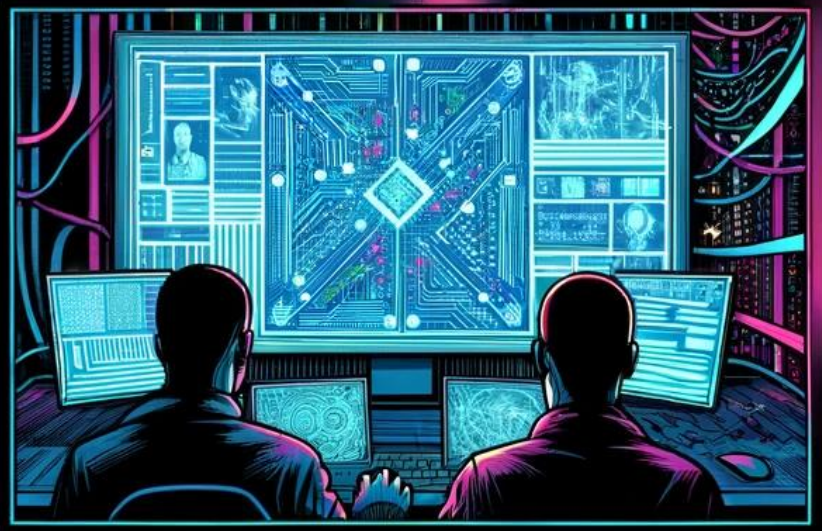
Analysis for targeting this individual: The email reveals a potential financial transaction of $3 million, which could be of interest to competitors or criminal organizations looking for financial gain. The individuals mentioned (Ben, Kay, and Mike) may also be of interest for further targeting and potential exploitation. However, without additional context or information about the transaction, it is difficult to determine the full value or potential targets.

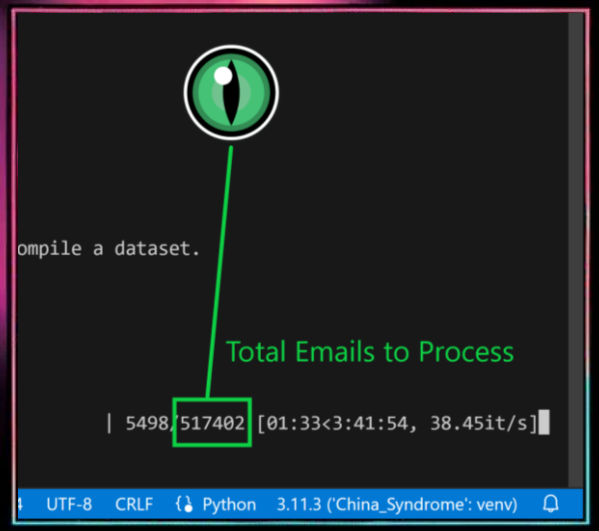
Raw Email Text

coordinating any needs and info on the closing (i.e. our receipt of FedRef #s) as Chris Booth is out of the office. Please contact her with any questions, and with the wire information. Her phone number is 713-345-7968, and her e-mail is rebecca.walker@enron.com. Please also include her on any additional e-mails. Thanks much. Ben "Keffer, John" on 04/26/2001 09:38:25 AM To: "Jeremiah A. DeBerry (E-mail)" cc: "C. Kay Mann (E-mail)" , "Michael Young (E-mail)" , "Ben F. Jacoby (E-mail)" , "Reuter, Marisa" Subject: Signature Pages and Wire Jerry-as we discussed, here are PDF files containing Enron's signature pages to the Letter Agreement and the two Change Orders. **As soon as you receive these, you will send to Ben, Kay and me your client's signature pages to the Letter Agreement and Guarantee, and Mike will begin the wire transfer of $3 million.** Mike will also send to Ben, Kay and me the Fed reference number as soon as he gets it. Finally, as agreed, Paul Hastings will assemble the documents for distribution to the parties. To that end, Kay, please send the 6 originals of the signature pages by overnight delivery to Jerry. Regards John L. Keffer King & Spalding 713 751 3255 713 751 3280 (fax) jkeffer@kslaw.com <> <> <> Confidentiality Notice This message is being sent by or on behalf of a lawyer. It is intended exclusively for the individual or entity to which it is addressed. This communication may contain information that is proprietary, privileged or confidential or otherwise legally exempt from disclosure. If you are not the named addressee, you are not authorized to read, print, retain, copy or disseminate this message or any part of it. If you have received this message in

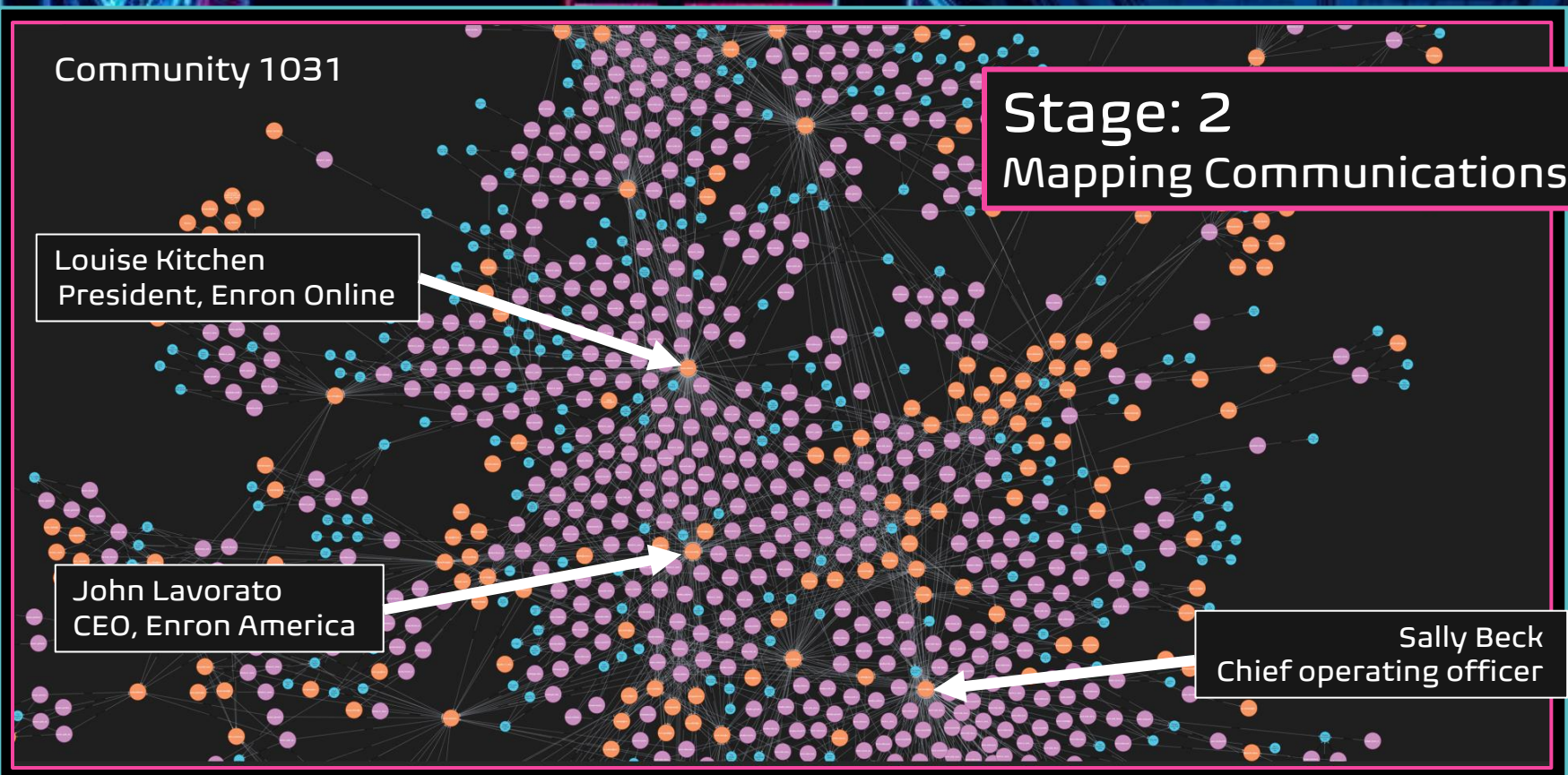
```python
LAW_Tokens = ["contract", "schedule", "agreement", "compliance", "regulation", "litigation", "patent", "trademark
embeddings = {}
with open('C:/Models/China_Syndrome/Data/crawl-300d-2M.vec', 'r', encoding='utf-8') as f:
    next(f)
    for line in f:
        values = line.split()
        word = values[0]
        vector = np.array([float(x) for x in values[1:]])
        embeddings[word] = vector

def get_word_vector(word, embeddings_dict, dimension=300):
    return embeddings_dict.get(word, np.zeros(dimension))

def get_embedding_centroid(words, embeddings_dict):
    valid_embeddings = [get_word_vector(word, embeddings_dict) for word in words]
    centroid = np.mean(valid_embeddings, axis=0)
    return centroid

law_centroid = get_embedding_centroid(LAW_Tokens, embeddings)

def get_confidence(entity_text, embeddings, law_centroid):
    placeholder_words = entity_text.split()
    sentence_centroid = get_embedding_centroid(placeholder_words, embeddings)
    cos_sim = cosine_similarity([law_centroid], [sentence_centroid])
    return cos_sim[0][0]
```

Stage: 1
NER & Custom Embeddings






```python
nlp = spacy.load("en_core_web_lg")
def get_entities(doc_dict, filter_list, all_emails_dict):
    ner_results = {}
    for doc_name, doc_content in tqdm.tqdm(doc_dict.items(), desc="Searching for espionage materials"):
        try:
            cleaned_content = doc_content.replace("=01,", "'").replace("=20", "").replace("= ", "").replace("=09"
            content_for_checking = cleaned_content.lower()
            excluded_keywords = ["wall street", "bloomberg", "dow jones", "trading", "invest", "tickets", "hotel"
            if any(keyword in content_for_checking for keyword in excluded_keywords):
                continue
            doc_nlp = nlp(cleaned_content)
            entities = []
            for ent in doc_nlp.ents:
                if ent.label_ in filter_list:
                    sentence = ent.sent.text
                    co_sine = get_confidence(sentence, embeddings, law_centroid)
                    entities.append({
                        "entity_label": ent.label_,
                        "entity_text": ent.text,
                        "Confidence": co_sine,
                        "Sentence": sentence
                        })
            if entities:
                ner_results = {
```




```
graph_db_instance.project_main_dataset(graph_name)
graph_db_instance.detect_communities(graph_name)
graph_db_instance.run_pagerank(graph_name)
graph_db_instance.calculate_degree_centrality(graph_name)
graph_db_instance.drop_projection(graph_name)
```

```python
def evaluate_entities_and_sentences(graph_db_instance, tokenizer, model):
    top_communities_entities = graph_db_instance.get_top_communities_and_entities()
    seen_sentences = set()
    for community, entities in tqdm.tqdm(top_communities_entities, desc="Analyzing entity communications and buil
        for entity in entities:
            entity_value = entity['value']
            top_sentences = graph_db_instance.get_top_sentences_for_entity(entity_value)
            for sentence_record in top_sentences:
                sentence_text = str(sentence_record['Text']).replace('"', '')
                sentence_hash = hash_sentence(sentence_text)
                if sentence_hash in seen_sentences:
                    continue
                seen_sentences.add(sentence_hash)
                prompt = f"Assume the role of a sophisticated state-sponsored adversary who has recently acquired
                response, elapsed_time, _ = mixtral(prompt, tokenizer, model)
                response_lower = str(response).lower()
                rate_value = 0  # Default to 0 (unknown)
                if "rate value level: high" in response_lower:
                    rate_value = 3
                elif "rate value level: medium" in response_lower:
                    rate_value = 2
                elif "rate value level: low" in response_lower:
                    rate_value = 1
                graph_db_instance.create_analysis_node(entity_value, sentence_text, response, rate_value)
```

Stage: 3
Large Language Model Analysis

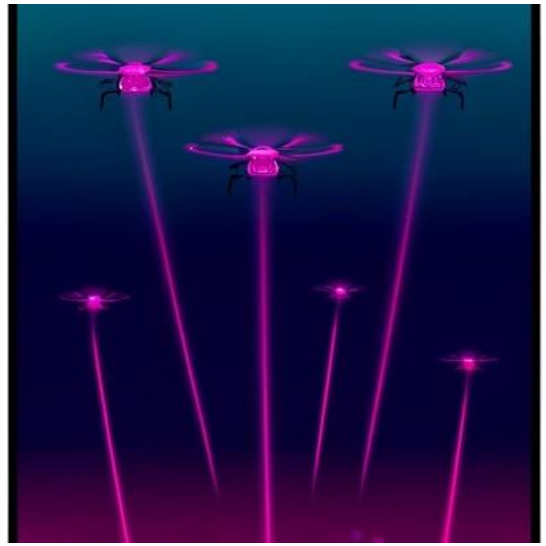
```
emails$ match (a:Analysis)--(s:Sentence)--(d:Document) where a.rateValue >1 r
```

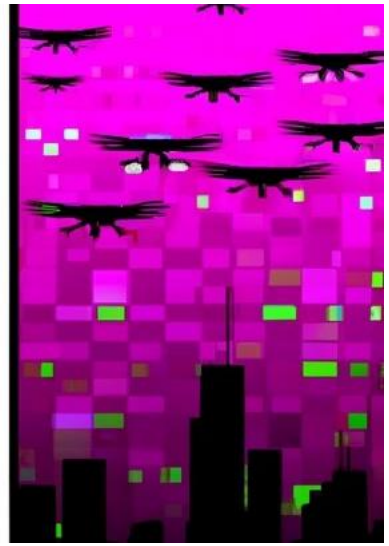
Table | Text | Code

count(d)
1305

Total Emails

Started streaming 1 records in less than 1 ms and completed after 3 ms.

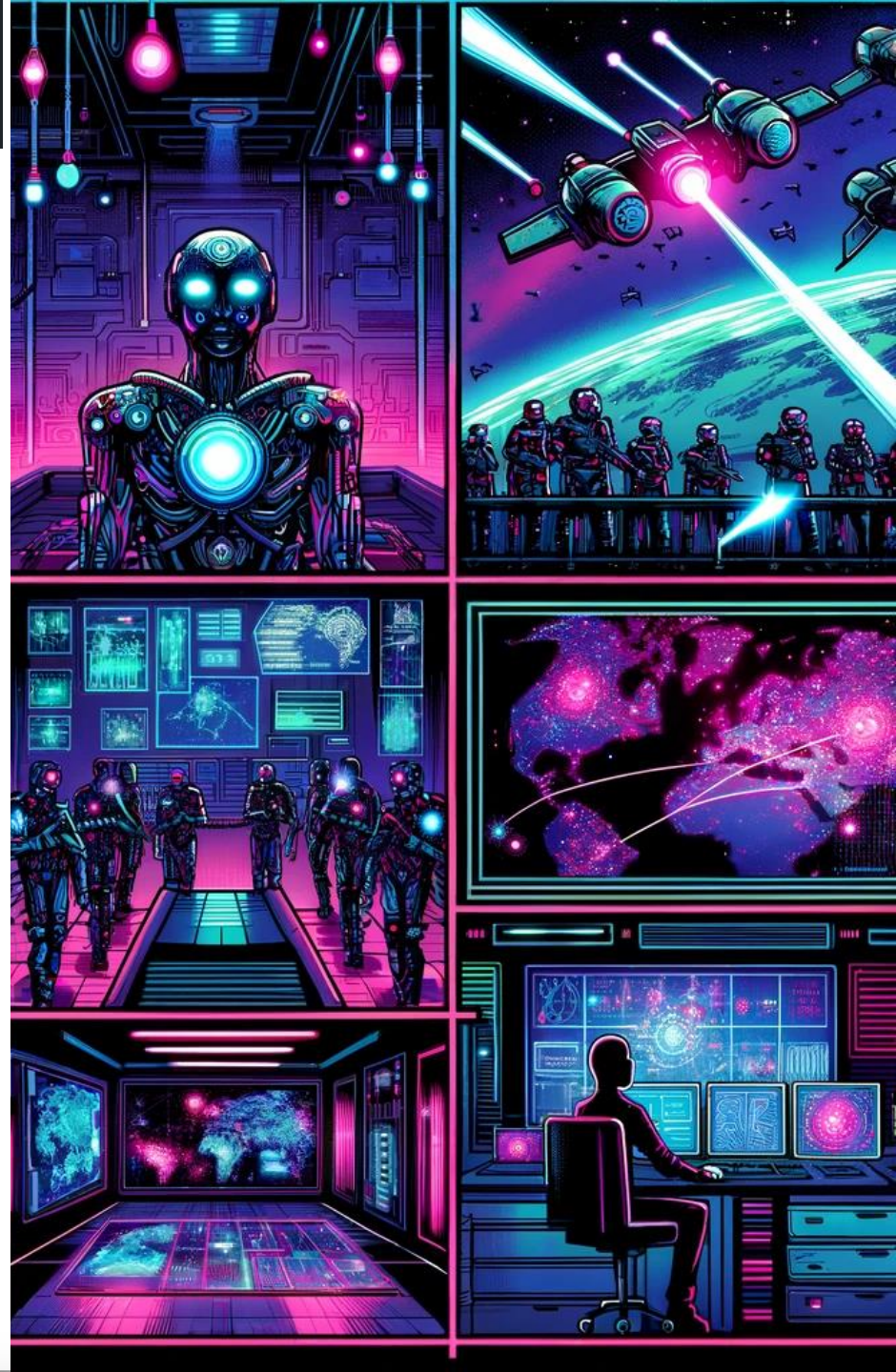


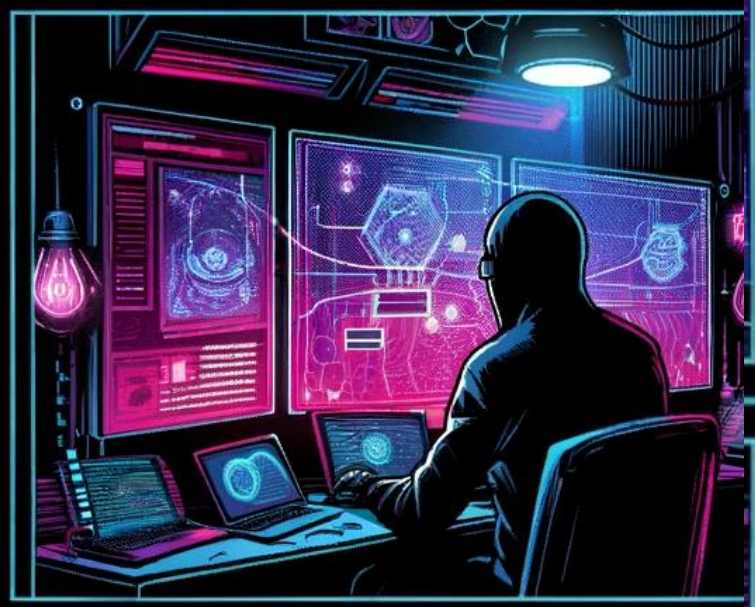


TI Cyber Hunter
Opensource Tool Drop (PoC)

graph.py	page # for multi-page docs	last week
main.py	page # for multi-page docs	last week
models.py	clean up	last week
readme.md	working on readme	last week

README





Closing Thoughts

- Cyber Offense / Defense Equilibrium
- Adversary AI Adoption
- Defender AI Adoption