# Phase I: Setup & Recon

Phase I is foundational to the entire BLACKICE offensive AI framework. The success of all following stages depends heavily on the efficiency, stealth, and comprehensiveness of this reconnaissance and setup phase. This stage is designed to identify, map, and assess the full breadth of a target’s digital footprint using both active and passive OSINT tactics, AI-assisted reconnaissance tools, and sophisticated threat modeling techniques.

## Objectives

1. Collect actionable intelligence on target infrastructure, personnel, domains, subdomains, and technologies in use.  
2. Establish stealth infrastructure (Command-and-Control servers, anonymized proxies, AI-assisted monitoring agents).  
3. Map out attack surface using graph analytics and threat topology generation.

## Tools & Resources

- Maltego CE/XL (Graph-based OSINT mapping)  
- SpiderFoot HX (Automated Recon)  
- amass (Subdomain enumeration)  
- Shodan + ZoomEye API  
- recon-ng (Reconnaissance framework)  
- Censys Search  
- DNSDumpster  
- Python3, Jupyter Notebooks  
- AI Toolkit: GPT-based scraping agents, LangChain, Haystack for LLM-enhanced intel parsing  
- CloudLab/AWS/Oracle Cloud (for spinning anonymous VMs)

## Code Snippets: Stealth Recon Infrastructure

Python3 code for setting up C2 and anonymized rotating proxies for passive recon:

import requests  
import time  
import random  
  
proxies = [  
 "http://123.45.67.89:8080",  
 "http://98.76.54.32:3128",  
 "http://11.22.33.44:8000"  
]  
  
def rotate\_proxy(url):  
 for i in range(5):  
 proxy = random.choice(proxies)  
 print(f"Using proxy: {proxy}")  
 try:  
 response = requests.get(url, proxies={"http": proxy, "https": proxy}, timeout=5)  
 print(f"Status: {response.status\_code}, Length: {len(response.text)}")  
 except Exception as e:  
 print(f"Request failed: {e}")  
 time.sleep(3)  
  
rotate\_proxy("http://targetdomain.com")

## AI-Powered Subdomain Enumeration using amass + GPT

Amass can be enhanced with LLMs to identify patterns and prioritize high-value targets. Below is a simple integration concept using Python subprocesses and LangChain to prioritize discovered subdomains.

# amass enum -d target.com -o subdomains.txt  
import subprocess  
from langchain.llms import OpenAI  
  
llm = OpenAI(temperature=0.2)  
  
with open("subdomains.txt") as f:  
 subs = f.read().splitlines()  
  
query = "Classify the following subdomains from most to least valuable for a cyberattack:  
" + "  
".join(subs)  
response = llm(query)  
print(response)