Spotter: The Discord Security Bot

Project Proposal

Henry Geng, Jenny Pan, Shu Yang, Ye Yu

Discord's decentralized server model, reliance on user-generated bots for functionality, and Content Delivery Network (CDN) infrastructure for file sharing enable attackers to weaponize the platform's own core features to distribute malware at scale. Unlike other social platforms where content flows through centralized channels, Discord's ecosystem is community-based and built on trusting strangers online. This unique phenomenon exacerbates security vulnerabilities where malicious actors employ social engineering tactics and embed malicious URLs to phish users, steal data, and install Remote Access Trojans (RATs). Large, public discord servers often face significant challenges in moderating scams, phishing attempts, impersonation, and other forms of social engineering. Human moderators often struggle to keep up with message volume, allowing malicious content to persist long enough to harm users. Existing Discord moderation tools offer basic filtering but lack robust, automated scam-detection capabilities tailored to evolving social-engineering techniques.

The primary delivery mechanism for malware occurs through malicious URL links embedded in messages and invite links disguised within trusted Discord interactions. No existing solution comprehensively addresses these attack vectors by combining automated social engineering detection with real-time URL threat analysis. Our project tackles this problem by creating an accessible bot that automatically detects scam-like messaging patterns and flags malicious URLs before users can interact with them.

The expect output of this project should be a functional Discord bot that is capable of:

- Detecting suspicious, scam-like, or phishing-related content
- Deleting or flagging dangerous messages
- Warning or automatically moderating potentially harmful users
- Checking URLs using Google Safe Browsing or similar APIs
- Providing logs summarizing incidents detected

Week 1

- Research background on Discord scams and social-engineering techniques
- Finalize feature list and program architecture
- Implement core bot functionality (message listening, moderation actions, role permissions, logging)

Week 2

- Develop or integrate a scam-detection system (ML API or Custom approach)
- Add URL/phishing detection via Google Safe Browsing API
- Combine malicious behavior APIs with core bot functionality

Week 3

- Robust testing on a private server
- Final polish, bug fixes, and documentation
- Prepare final project report and demo

Team Responsibilities

Henry: Core bot implementation functionality
Jenny: Research, write the Problem/Significance section, and implement URL-checking functionality
Ye: Integration of scam-detection modules
Shu: Integration of modules with core, testing and polish

References for Research

This project proposal was developed with assistance from Claude (Anthropic), an AI assistant used for research organization, writing refinement, and technical ideation.

1. New Jersey Cybersecurity & Communications Integration Cell (NJCCIC). "Discord Invites Used as a Path to Malware." *Latest Alerts and Advisories*. NJCCIC. https://www.cyber.nj.gov/alerts-advisories/discord-invites-used-as-a-path-to-malware *Documents real-world cases of weaponized Discord invite links. Establishes invite-link attacks as a primary Discord-specific threat vector.*
2. "From Trust to Threat: Hijacked Discord Invites Used for Multi-Stage Malware Delivery." *Analyzes multi-stage attacks through hijacked Discord invites. Shows how attackers exploit community trust in Discord's invitation system.*
3. CYFIRMA. "Analysis of a Discord-Based Remote Access Trojan (RAT)." *CYFIRMA Research*. https://www.cyfirma.com/research/analysis-of-a-discord-based-remote-access-trojan-rat/ *Examines Discord's use as a command-and-control channel for RATs. Demonstrates how Discord's API enables persistent malware operations.*
4. Group-IB. "Remote Access Trojan Threats and Defense." *Group-IB Threat Intelligence*. https://www.group-ib.com/resources/threat-research/remote-access-trojan/ *Provides foundational knowledge on RAT capabilities and attack methods. Informs detection patterns for Discord-specific RAT distribution.*
5. Google Developers. "Safe Browsing API." *Google for Developers*. https://developers.google.com/safe-browsing *Technical documentation for implementing real-time URL threat detection. Essential for our bot's malicious link identification functionality.*