**Summary**

“BotHunter: Detecting Malware Infection Through IDS-Driven Dialog Correlation” is an academic study and experiment where Georgia Institute of Technology along with other participants such as SRI International present a new type of network monitoring strategy which integrates infection recognition and coordination dialog analysis. Over the last decade, malware has become the source of most botnet initiated denial of service attacks. The typical stages of malware infection are as follows: inbound scanning, exploit usage, egg downloading, outbound bot coordination dialog, and outbound attack propagation. However, botnets are different from most other malware in that they seek to establish a line of communication to a command and control channel from which they can be directed and updated. Intrusion detection systems typically only monitor incoming packets. BotHunter utilizes a multi-threat attack recognition strategy – SNORT – which involves three main components: malware/botnet rule based detection, SLADE (payload analysis) and SCADE (incoming and outgoing traffic analysis). These three components are analyzed by the Java BotHunter correlator algorithm which produces a bot infection profile consistent of relevant information such as confidence score, victim IP, attacker IP, etc. SLADE analyzes the payload for metrics to include payload size deviation. The correlator algorithm determines that a potential botnet infection has crossed the threshold.

The strengths of this paper are the initial experiments. The authors tested various setups in Virtual Machine environments using different botnet malware and provided results which were impressive for an experimental system. They tested with known botnet malware softwtare . This proof of concept description and testing methodology were clear, concise, and relatively sound.

The weaknesses of this paper were very minor. I would have liked to see the effect of running existing market anti-virus solutions on the detection rates. For example, would BotHunter conflict or complement existing software on the market? Would they work better together or worse together?

**Suggestions for Improvement**

There are not many novel suggestions for improvement for this paper. The authors already did the community a favor by highlighting the areas that need emphasis and improvement. It was noted that future threats and adversaries would evolve so BotHunter must also evolve with that threat. It is also conceivable that future threats would seek to evade detection algorithms used by BotHunter by adopting more stealthy scanning techniques. Future threats may also target BotHunter itself if BotHunter becomes more mainstream. Attacks like these would include deleting BotHunter log files. Countermeasures should be developed to counter these future threats. BotHunter could also be improved by ensuring compatibility with more platforms and environments as well as add more state logic besides the existing three major components.

**Future Research**

Future research should focus on the how the community aggregates malware and botnet signatures and profiles. The main challenge is being able to share information and data in an understandable and widely accepted forum. A continuous effort must be maintained to study malware and intrusion sets as well as cover new intrusion sets.