Rami Naser

ITMD 469 Go Programming

The Use of GO in cyber security.

Go is an incredibly useful tool for defensive cybersecurity. Its standard libraries allow the creation of network security tools, encryption and decryption, file manipulation, and even automation. Go an incredible tool for network scanning and vulnerability testing. It has concurrency and multithreading capabilities that can run many processes at the same time, effectively dividing the time it would take to do those tasks sequentially by the number of available threads on your CPU. This speeds things up the same way it did when intel went from single core processors to modern Xeon processors that can do over 160 tasks at once. What's even better is all these capabilities are built in even the ability to do this with networking commands and port scanning. Now yes tools that do these things already exist, so why might I want to make a new from scratch? Portability and compatibility. Go will run on any modern computer with speed and efficiency. I can essentially create one tool, copy it to a flash drive and use it on anything I can get my hands on even if it doesn't have a lot of resources. Coupled with Go's ability for file manipulation and built in cryptography I could very easily run something like a docker container that always scans my business network for unrecognized services, devices, mac addresses, or anything else I want and collect those logs. Then comes the trivial task of creating logs and even encrypting them for added security. From my own experience I know this would be gold in the hands of a robust IDS or SIEM tool that every business should have. Better yet this replaces some expensive add on functionality of those tools for the cost of a little docker container.

The Dark Side of Golang

While Go does have many uses for the blue team but it's not all sunshine and rainbows. Plenty of threat actors use Go for the same reasons we do and the same reasons as the good guys like security analysts. However, one thing that makes it especially aggravating are it's easy to obfuscate malicious code since the built in Libraries are incredibly powerful and custom named methods and variables make it where it's really only useful to look at main. At the same time GO is very efficient, has a light footprint, and does not care which OS you are using it will run on them all.

Go can be an incredibly useful tool for any threat actors that need to make a quick buck and want to avoid detection. According to CrowdStrike, most malware written in the GO programming language is to use the infected computer's resources to mine crypto currency. They claim that this is because of some of the same aspects of GO that we praise as being useful features or make it worth learning. It's incredibly efficient, up to "40 times faster than optimized python code", and it also is pretty much operating system agnostic, meaning it doesn't care if you are using Linux, mac, or windows. Malware written in GO is also very hard to comb through and analyze. All of these features make it incredibly desirably not only for everyday threat actors like that gifted kid in his mom's basement but also for Advanced persistent threats. These are the groups like Lazarus out of North Korea. It's a state funded cyber terrorist organization that carries out attacks by hacking whatever strategic targets they can think of. But they are also isolationists with a miserable economy but want to fund things like missile programs and statues of their glorious leader, so they steal that money. One way they do this is with ransomware and mining viruses like the one's CrowdStrike reports are becoming more and more popular. The money they steal from these ransomware attacks and mining viruses are also directly linked to funding heinous crimes, kind of like the Taliban selling heroin. The same features of go that make it so easy for us to use and so useful for lightweight and efficient programming no matter the system make it the premier choice for cyber terrorists, APTs, and other threat actors to create malware.

Sources

Moss, R. (2025, March 30). *Building a Network Vulnerability Scanner with Go — SitePoint*. Sitepoint.com. https://www.sitepoint.com/building-a-network-vulnerability-scanner-with-go/

Anmol Maurya. (2025). *Financial Motivation Drives Golang Malware Adoption | CrowdStrike*. Crowdstrike.com. https://www.crowdstrike.com/en-us/blog/financial-motivation-drives-golang-malware-adoption/

MrEhAcKeR. (2025, February 8). *Automating Cybersecurity with Golang - MrEhAcKeR - Medium*. Medium. https://medium.com/%40mazin.ahmed.business/automating-cybersecurity-with-golang-72502972a02b