Paper instructions: Background: The main defense against malware continues to be antivirus software, which uses a combination of signatures and heuristic rules to detect malware infections. But where do the signatures come from? Security companies collect and inspect new malware samples to identify new ones that are interesting enough for thorough analysis. Expert analysts use a variety of tools to reverse engineer and understand a suspected binary. Dynamic analysis involves execution of a suspected binary or executable to learn about its possibly malicious behavior. Generally, dynamic analysis looks for suspicious behavior with regards to the following: • Actions on the machine where it is running, e.g., buffer overflows, file changes; • Network traffic, e.g., communications with C&C (communications and control) servers; • Attempts to self-replicate Dynamic analysis can be complicated when malware creators design malware to change its behavior if it detects the presence of a virtual machine. Clearly, execution should be done in a restricted environment like a sandbox to protect the network and other machines. There are obvious costs in computing resources and execution time. Thus, it is not feasible to carry out dynamic analysis for every suspected binary. In addition, a high level of technical expertise is needed to understand the results of dynamic analysis. Dynamic analysis, as well as static analysis, is much like detective work. Please address the following: • Summarize static analysis attempts to learn from a suspected binary. • What are the limitations of static analysis, or in other words, why is dynamic analysis needed? • Give an example of program behavior that can be learned only through dynamic analysis and not static analysis. • Summarize the risks of dynamic analysis. • Give an example of an included package in Cuckoo Sandbox?