PLAN OF WORK & METHODOLOGY

1.4.1 Selection of various techniques of an IDS depends on the TCP/IP,
   Based on the information obtained from the literature review, network layers (AIDS, TIDS, NIDS, and LIDS) in improving both system performance and scalability in Deep Packet Inspection (DPI).

1.4.2 Design of proper encoding using Integrated TCP/IP Protocol Software.
   To maximize the success of the functional approach for detecting software vulnerabilities in network protocol implementations, we generalize our methodology as a black box approach with fault injection tests designed by mutating message, syntax, content and sequence under specific protocol running scenario.

1.4.3. Selection of proper High Throughput Parallel Architecture.
   The contest allowed for a design submission to be on any platform using any combination of hardware and software. To make the contest platform agnostic, the Ethernet stream is simulated as a network buffer with packet data in memory. The contest provided an x86 C++ based reference design, running this on a high powered workstation showed a throughput of less than 10 Mbit/sec under High Throughput Parallel Architecture.

1.4.4 Design High Speed Pattern Matching for Deep Packet Inspection (DPI)
   In our proposed single-character string matching method, the number of TCAM entries required to find a string is $c$, where $c$ is the number of characters included in the string. This method does not require memory for state transition. Instead, a control circuit and an index encoder are required

1.4.5 Development of Efficient Memory Utilization for DPI
   Deep Packet Inspection (DPI) refers to examining both packet header and payload to look for predefined patterns, which is essential for network security, intrusion detection and content aware switch.

1.4.6 Development of Fast and Memory Efficient Traffic Classification
   Traffic classification is important to many network applications, such as network monitoring. The classic way to identify flows, e.g., examining the port numbers in the packet headers, becomes ineffective
1.4.7 Implementation of Traffic-Aware Frequent Element Matching Algorithm for DPI

Network devices are increasingly using packet content for processing incoming or outgoing packets. Many pattern matching algorithms have been proposed to improve packet matching throughput. Most of them are, however, independent of traffic pattern and may end up with longer match time against actual traffic.

1.4.8 Analysis of the results and conclusion:

The results obtained from the execution of the algorithm will be analyzed and compared with the existing system. And expected results will be submitted as a report.