CERBERUS

About Cerberus

Cerberus is a complete solution used to monitor neighbourhoods through Wireless Sensor Networks (WSN). It consists of code loaded on motes, a simulator system and a stand-alone application. Readings detected by the Wireless Sensor Network is routed to the Cerberus application from where it is processed. Cerberus allows the user to configure exactly how it should interpret packets received from the Wireless Sensor Network and through a sophisticated rule system the user can control how he/she should be notified and when. Through Cerberus a thorough, hard to breach self-healing security system wrapped in an efficient graphical user interface is finally introduced for all to use.

The Project

Cerberus is more than just a security system. As a project it involves the research into a new technology and the creation of a practical implementation for that technology. Cerberus illustrates that Wireless Sensor Networks is a viable solution to various problems like security, environment monitoring and ad hoc monitoring.

Because of the use of Wireless Sensor Networks, Cerberus allows for rapid deployment and ad hoc sensing of areas. It is very difficult to disrupt and will one day be cheaper than its peers. It is completely dynamic in its addition of motes and removal and is perfect for temporary solutions.

Components

TinyViz Plugin allows the user to simulate the functioning of the *Cerberus* system without the need to deploy actual motes. It is an excellent tool for managing changes in the system.

XML Plan Application allows the user to create a floor plan of the area where the motes are to be placed. This floor plan can be imported by Cerberus and Tinyviz.

Cerberus Server allows multiple clients to connect to the same Wireless Sensor Network. It decreases the work load on the client and lends greater control to the system in its whole.

GANG OF FIVE

espresso

CERBERUS SECURITY SYSTEM

Components

Web Interface allows the user to connect to his/her Cerberus system from any computer with internet access.

Rule Engine provides the user with an interface to create sophisticated event/action rules. These rules are processed as readings are sensed and the desired actions carried out.

Remote Mote Control through packet injection allows the user to control the functioning of the motes without modifying the physical motes or the code running on it.

Logging Engine keeps track all values sensed and stores it for analysis