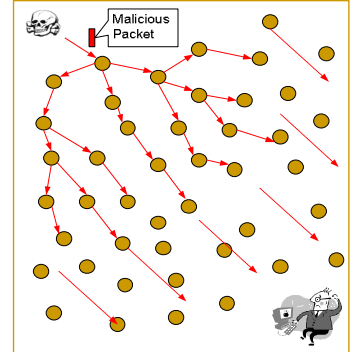


## Motivation

- Internet worms replicate themselves to infect and propagate among as many computers as possible in a very short time, by exploiting e.g. buffer-overflow vulnerabilities
- Is worm attack possible in sensor networks? Yes
  - Simple hardware architecture and OSES provide little protection
  - Sensors in the same network have homogeneous hardware and software

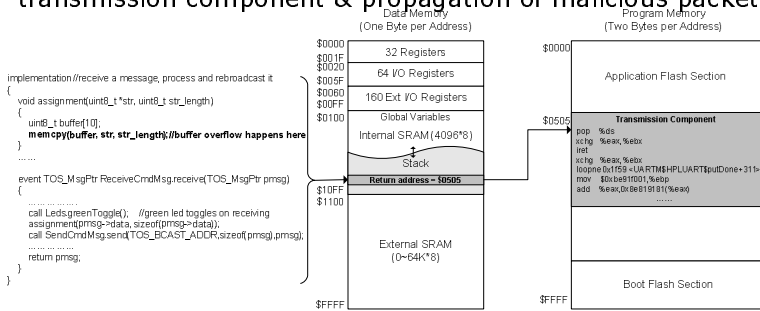
## Sensor Worm

- Crafted messages that exploits software vulnerabilities of sensors, causing sensors to crash or taking control of sensors
- Worst case:
  - A single message to compromise the entire network



## Contribution I

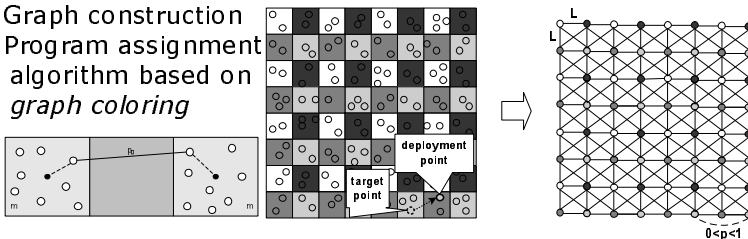
- We illustrate the feasibility of launching sensor worms through *trial experiments* on Mica2 motes
- Buffer overflow may result in transfer of program flow to transmission component & propagation of malicious packets



## Contribution II

- In the spirit of survivability through heterogeneity philosophy, we explore *software diversity* for sensor worm defense

- Graph construction
- Program assignment algorithm based on *graph coloring*



- Impact of sensor deployment error: deployment points are modeled by *two-dimensional normal distribution* with target points as mean

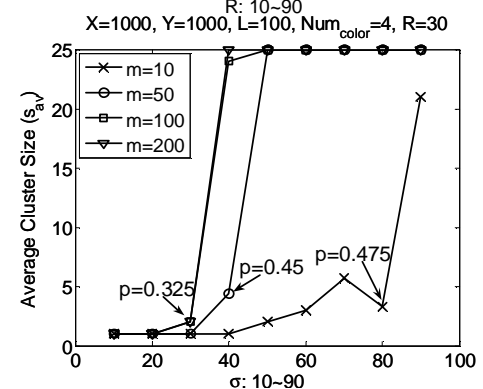
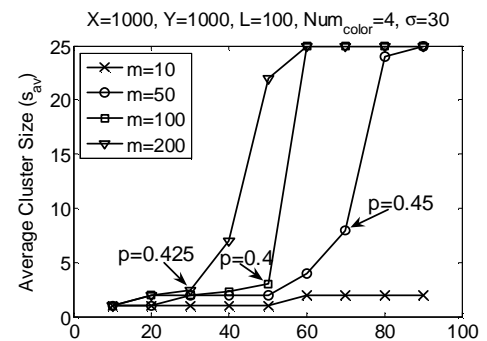
Probability that two nodes from neighboring cells with the same color are connected

$$p_0 = \int_{x=0}^X \int_{y=0}^Y \frac{P_{n2}}{2\pi\sigma^2} e^{-[(x-x_1)^2 + (y-y_1)^2]/2\sigma^2} dx dy$$

Probability that two neighboring cells with same color  $p = 1 - (1 - p_0)^m$

## Performance Evaluation

- Average cluster size based on *percolation theory*



- Compare worm containment with other schemes

