Misuse Detection and Prevention in Wireless Ad-hoc Networks

Reuven Gevaryahu  J. Robert Yaros
Faculty Advisor: Saswati Sarkar

Intrusion Detection Systems create an alert whenever malicious packet discovered, but do not stop the packet from reaching its destinations. Using promiscuous mode, a single node can detect malicious packets from a large number of senders. To minimize resource consumption, the goal of IDS becomes running the system on the fewest number of nodes possible, yet still guarantee the transmission of every packet will be heard by at least one node running detection software. This project compares two node selection algorithms: Dominating Set (DS) which attempts to choose nodes with highest degree and Random Probability (RP) where each node activates its detection software with certain probability.

**DS Algorithm**
1. Start Active
2. Send Degree to Neighbors
3. Tell All but Highest Degree Neighbor to Disable. (Ties have special cases)
4. Also tell Highest Degree Neighbor to Disable if Degree is lower than Self
5. Disable if all Neighbors say to Disable
6. Stay Active if have no Neighbors

**Test Bed**
Testing of IDS and IPS implementations was performed on a test bed network of PDAs and a variety of Laptops.

**IPS Implementation**
- User Space
  - Snort Attack Detection Software
  - Snort_Inline Attack Prevention Software
  - libpcap/libipq Packet Capturing/Dropping
  - Netcat Attack Creation
  - Perl & Bash Scripting

The IPS implementation consists of a version of Snort Inline modified to accept input packets via shared memory, and a modified version of the AODV routing software that would send packets to the modified version of snort, based on information about the packet's route.

**Testing**
Results show that First is most efficient for the network, allowing more data to be transferred from the legitimate sender to the destination. Last has the benefit of being invulnerable to spoofing, but at a cost of allowing the packet to congest the network all the way until the last hop. Both has the advantages of First and Last, with the disadvantage of an additional scanning delay.

Resource usage is the same in aggregate for First, Last and Both because all nodes run the AODV software and load Snort Inline. Both has slightly higher CPU usage. Random mode saves memory on aggregate, in exchange for the loss of guaranteed attack prevention.

Special Thanks:
- Farooq Anjum (Adjunct Professor)
- Dhanant Subhadrabandhu (PhD Student)

Senior Project Poster Day 2005, CIS Dept. University of Pennsylvania