Routing dilemma

- Conventional routing protocols are optimized for performance, not security
- Adding security measures such as router authentication is expensive and performance will suffer
- Security measures don’t scale very well
- Classical performance / security trade-off

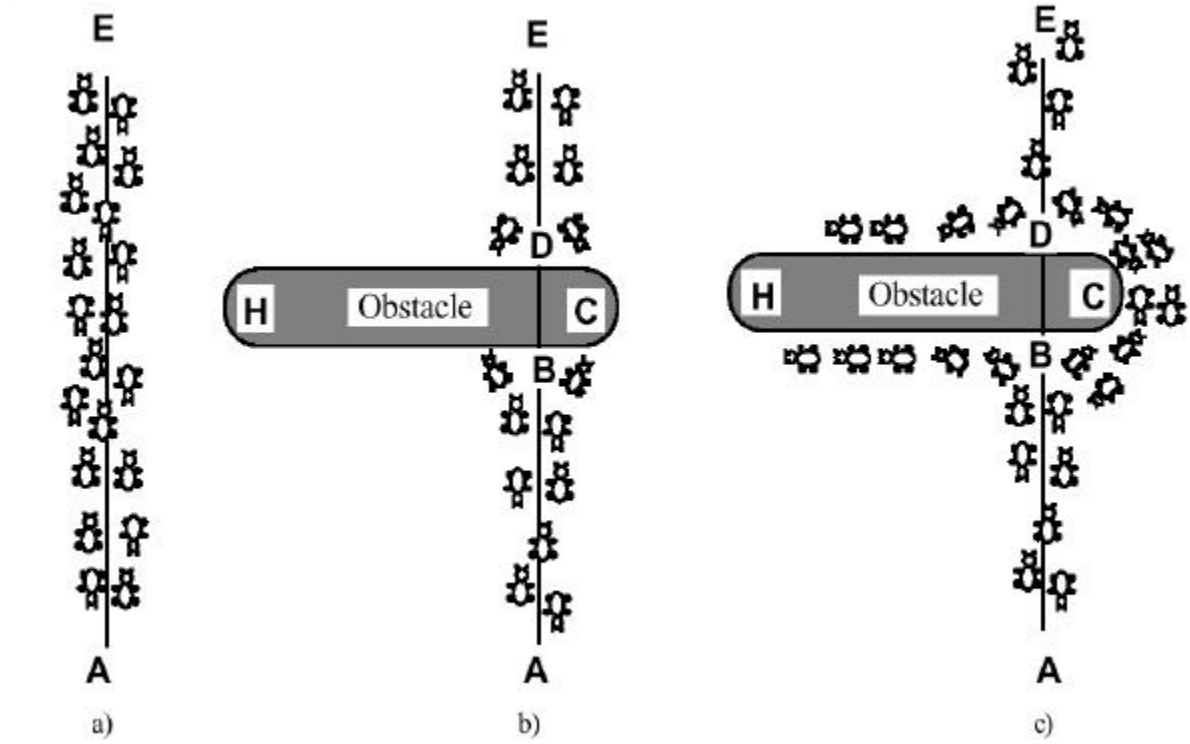
Solution

- AntNet routing delivers excellent performance
- A few modifications of the original work provide easy but yet effective security controls
- Performance is retained as no expensive cryptographic functions are used, only hashing
- Low extra overhead in network traffic for added security

Ant System Basics

- Based on the behavior of real ants
- *Pheromone trails* are used to communicate information about individuals
- The more ants follow a trail, the more attractive that trail becomes for being followed (positive feedback)
- Shorter routes are emphasized more strongly, favoring shortest paths

Example



- Ants follow a path between points A and E.
- An obstacle is interposed; ants can choose to go around it on either side with equal probability.
- On the shorter path more pheromone is laid down.

Adopted from M. Dorigo et. al., “The Ant System: Optimization by a colony of cooperating agents”, IEEE Trans on Systems, Man, and Cybernetics, Vol. 26, No. 1, 1996, pp.1-13

AntNet Routing

- Based on work by M. Dorigo and G. DiCaro
- At certain time intervals, network nodes send out ants to selected destination nodes
- Two types of ants: forward and backward ants
- Routing tables contain probabilistic information about the “goodness” for choosing a node *n* for a particular destination *d*

Forward Ants

- Try to find a path from source *s* to destination *d*
- Keep track of path taken so far on their own stack
- Use probabilistic routing table entries to make decision about next hop
- Expire after a certain number of hops (TTL)
- Treated as normal data packets by routers

Backward Ants

- If destination is reached by forward ant, the ant becomes a backward ant
- Trace back the path stored on the stack
- Modify probabilities in routing table at each router, emphasizing the current path and possibly sub-paths
- Are forwarded with a high priority at routers

Performance

- Forward ants can be used to collect topology information such as queuing times
- AntNet routing outperforms common routing algorithms concerning packet delay while achieving similar throughput results
- AntNet has a slightly higher consumption of network resources than other algorithms, but this is by far outweighed by the better performance

Problems with AntNet

- The information stored on the stack is not protected from modification
- No method of associating a backward ant with a valid forward ant
- Attacks on AntNet include:
  - Flood the network with forward ants that already contain bogus path information
  - Send out false backward ants to advertise paths to go through a particular node

Securing AntNet

- Each router that receives a forward ant can calculate a cryptographic hash over the content of a packet appended with a key-string
- The hash is passed along with the ant as a *token*
- When a backward ant arrives at a router, the content and the key are used again to calculate a hash, which is then compared with the token

Implications

- Routers can be certain that the information advertised by the backward ant is correct
- The backward ant must have passed the router as a forward ant before
- Key management is completely autonomous for each router
- Only fast hashing but no expensive cryptographic computations are used

Open issues

- Increased security through probabilistic routing?
- The extra overhead (packet lengths) and performance implications have yet to be determined for Secure AntNet routing
- No solution yet for the following attacks:
  - Drop ant packets but let other traffic through
  - There is no method to ensure that an ant had actually visited its original destination
- Possible solution: peer monitoring