

# Honeypots & Honeynets

# Motivation

- Worms use random addresses sometimes
  - While scanning for vulnerable machines
  - While propagating
- Not all addresses are assigned
  - Packets headed for unassigned addresses indicate attacks!!

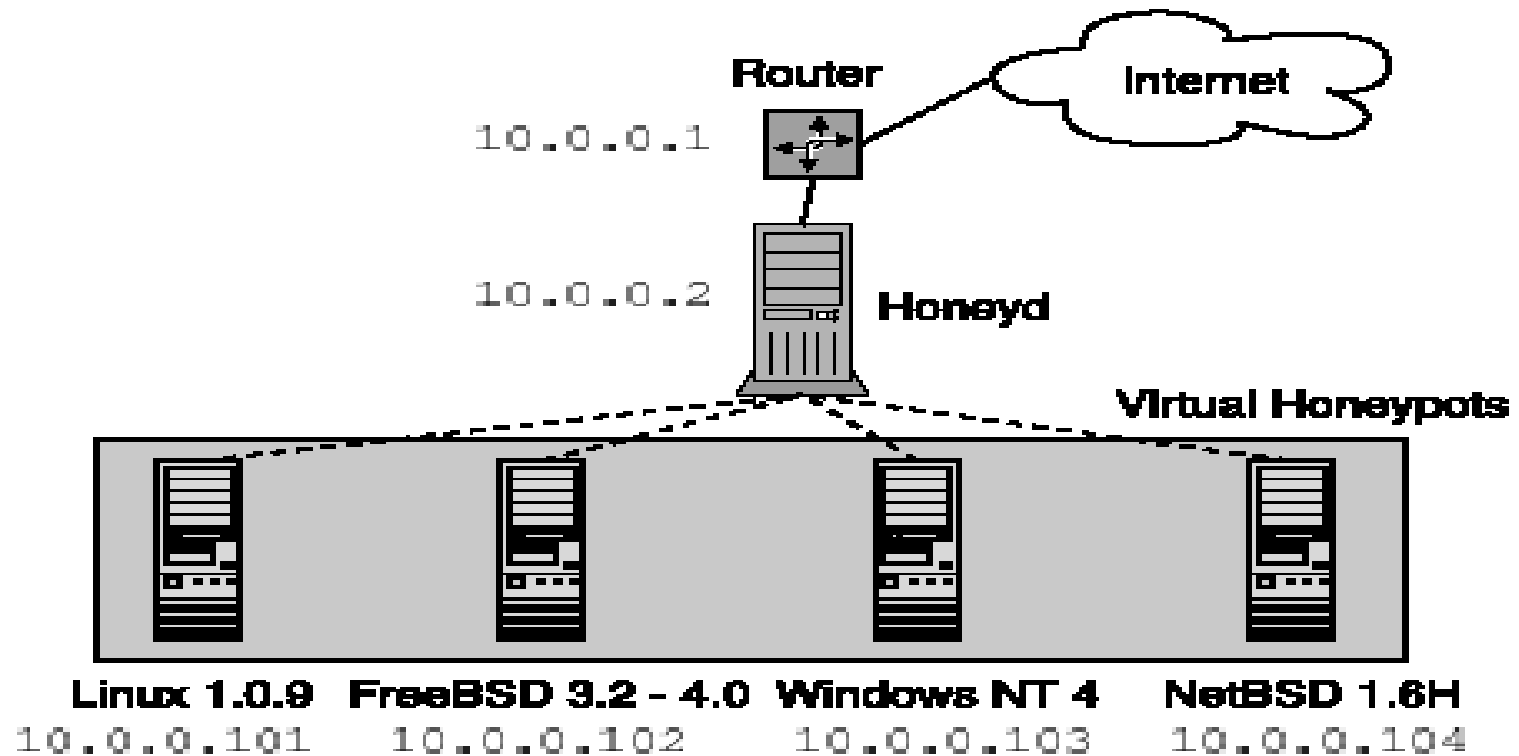
# Honeypot

- A machine set up to behave like a normal machine with an intent to track attacks
  - Could be a real machine with all services
  - Virtual machines with limited functions such as network stack
- Honeyynet: A network of honeypots

# Routing packets to honeypots

- Machine B pretends to be  $V_1, V_2 \dots V_n$
- Set up routing tables to direct traffic to  $\{V\}$  to B
- Use ARP -let B respond to ARP queries of  $\{V\}$  with its own MAC address
- Use tunneling to send packets to B

# Honeyd





# Network decoys

- Set up a fake server pretending to be real
- Use the services on this server to log human/worm attack activity
- Decoy can be compromised

# Unassigned addresses

- Packets to these addresses can be a good signal of worm propagation or attacks
- Can use the payload of packets for signature generation
- Deploy multiple honeypots across the globe
  - Same payload observed in different places reinforces the detection

# Other ideas(1)

- Look for packets with TCP Resets
- These also indicate attempts to connect to services that are not open
- Can detect scanning worms
- Weaver, Staniford, Paxson, "Very fast containment of scanning worms"

# Other ideas(2)

- Worms tend to use random IP addresses without contacting DNS servers
- Look at machines sending lots of packets without contacting DNS servers
  - Probably infected

# Other ideas (3)

- Can control worm propagation by limiting the number of new connections initiated by a machine
- Humans can't tell the difference
- Worms will be slowed down
- Will allow more time for detection
- Williamson et al, "Virus Throttling"

# References

- [1] N. Provos, "A virtual honeypot framework", Usenix Security Symposium, Aug. 2004