Monkey.org Developments

Honeypot Background

Honeyd Resources

Main - News - Forums
Download Releases
General Information [Mirror]
Frequently Asked Questions
Sample Configurations
Tools - Service Scripts
Blog - New
Links, Press, etc.
Mailing List Archive
Acknowledgments

Honeyd Research

Immunization Against Worms
Understanding Spam
Performance

Honeypot Resources

Honeypot Background
Honeypot Concepts

Provos Blog

Support my videos on Patreon!
Add your support on Patreon to help me create more videos. Your support will help with materials, rent as…
Forging the Finnish Spearhead from Rovaniemi, Marikkovaara: Part 3
My journey in recreating famous Finnish spearhead from Rovaniemi, Marikkovaara continues. In this episode, both the socket and the…
Forging a Wolf Tooth Spear: Part 2
Here is Part 2 of my new A Spear Born of Fire video series. My journey in forging the…

HEADLINES BY FEEDBURNER

Honeypot Books

Honeypot Background

A *honeypot* is as a closely monitored computing resource that we intend to be probed, attacked, or compromised. The value of a honeypot is determined by the information that we can obtain from it. Monitoring the data that enters and leaves a honeypot lets us gather information that is not available to NIDS. For example, we can log the key strokes of an interactive session even if encryption is used to protect the network traffic. To detect malicious behavior, NIDS require signatures of known attacks and often fail to detect compromises that were unknown at the time it was deployed. On the other hand, honeypots can detect vulnerabilities that are not yet understood.

Because a honeypot has no production value, any attempt to contact it is suspicious. Consequently, forensic analysis of data collected from honeypots is less likely to lead to false positives than data collected by NIDS.

Honeypots can run any operating system and any number of services. The configured services determine the vectors available to an adversary for compromising or probing the system.

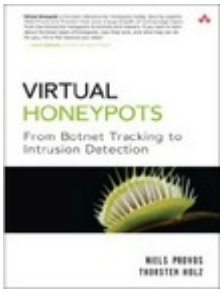
- A **high-interaction honeypot** simulates all aspects of an operating system.
- A **low-interaction honeypot** simulates only some parts, for example the network stack. This is what Honeyd does.

High-Interaction Honeypots

A high-interaction honeypot can be compromised completely, allowing an adversary to gain full access to the system and use it to launch further network attacks.

Low-Interaction Honeypots

In contrast, low-interaction honeypots simulate only services that cannot be exploited to get complete access to the honeypot. Low-interaction honeypots are more limited, but they are useful to gather information at a higher level, *e.g.*, learn about network probes or worm activity. They can also be used to analyze spammers or for active countermeasures against worms.



Virtual Honeypots.
Niels Provos and Thorsten Holz

Happy Hacking

Reduce wishlists
Leave a tip with PayPal

Support Honeyd

Search: Books
Keywords: Go
Search Amazon

We also differentiate between *physical* and *virtual* honeypots.

- A **physical honeypot** is a real machine on the network with its own IP address.
- A **virtual honeypot** is simulated by another machine that responds to network traffic sent to the virtual honeypot.

When gathering information about network attacks or probes, the number of deployed honeypots influences the amount and accuracy of the collected data. A good example is measuring the activity of HTTP based worms. We can identify these worms only after they complete a TCP handshake and send their payload. However, most of their connection requests will go unanswered because they contact randomly chosen IP addresses. A honeypot can capture the worm payload by configuring it to function as a web server. The more honeypots we deploy the more likely one of them is contacted by a worm.

Physical versus Virtual Honeypots

Physical honeypots are often high-interaction, so allowing the system to be compromised completely, they are expensive to install and maintain. For large address spaces, it is impractical or impossible to deploy a physical honeypot for each IP address. In that case, we need to deploy virtual honeypots.

You can find more information on Honeypots in Lance Spitzner's paper.
