

A Few Words About Attackers

- Experienced hackers that discover new vulnerabilities and write new exploits
 - Then share them on the Web or via IRC
- Script kiddies – download and deploy ready-made scripts
 - Inexperienced but in the majority
 - Use auto-rooter code to plug in any new exploit, then have it scan and break into machines using this new exploit
 - Mass-rooters do the same thing but probe different vulnerabilities

Honeypots

- Honeypot is an unsecured computer, with a goal to let the attackers in
 - So we can observe them! (research honeypots)
 - So we can protect our network by diverting attacks to honeypots (production honeypots)
- Unlike other defense measures it does not try to prevent any attack
 - We love attacks, especially new ones!
 - Honeypot acts like a canary in the mine
- The goal is simply to learn about attackers

Honeypot Types

- A honeypot can be:
 - **Emulated:** A single honeypot can emulate the whole network of computers with different OSs
 - Early honeypots were limited in a sense that they only emulated a first few steps in the compromise – attackers could not gain impression of success
 - **Real:** Real insecure machine augmented to log a lot of data
 - Gives attackers a real OS to interact with, they can do many more things
 - Uses firewall to stop further propagation of the attack code
 - Old machines
 - Especially built honeypots

3

Honeypot Types

- A honeypot can be:
 - **Production:** A scapegoat – alerts organization about a new compromise, slows down the attacker
 - **Research:** Gives the attackers impression that they have succeeded. Logs everything so we can learn about attackers.

4

How Do Honeypots Work?

- Any traffic a honeypot receives must be malicious (since honeypot runs no useful services for us)
- In research honeypots:
 - We need to save everything of importance
 - We need to give the attacker impression of complete control – along with outbound connection permissions
 - But we have to prevent him from compromising other machines

5

Honeypot Types

- A honeypot can be:
 - **Low-interaction:** Emulates variety of services, do not try to deceive attacker that compromise succeeded. Primary value is detection. These honeypots are easy to maintain and have a low level of risk.
 - **Medium-interaction:** Emulate a few services with more details so that the attacker would be lured to attempt the exploit. We can also partition the OS so to give the “look and feel” of a true OS but to quarantine actions. Complex and time-consuming, greater risk.
 - **High-interaction:** Dedicated OS representations. Must be properly set up to give the impression of total control but to prevent further spread. Highest risk.

6

HoneyPot Solutions

- BackOfficer Friendly
- Specter
- Honeyd
- ManTrap
- Honeynets

7

BackOfficer Friendly

- Low-interaction and free. Runs on Windows or Unix
- Designed as response to Black Orifice
 - It pretends to be a Black Orifice server – listens on the same port and emulates transactions
 - Logs attackers IP address and operations he tries to perform
- Also monitors some other services: FTP, Telnet, SMTP, HTTP, POP3, IMAP
- Does not forge particular packet fields, simply completes connection, then tears it down. It may attempt to generate some meaningful replies.
- Try it out: <http://www.nfr.com/resource/backOfficer.php>

8

Specter

- Low-interaction, commercial. Runs on some Windows.
- Emulates 7 services, 6 fixed and 1 customizable trap
 - Even vulnerabilities are emulated
 - It can emulate 13 different operating systems – one at a time but only at an application level
 - Captures attackers keystrokes and can even gather intelligence on the attacker
- Specter can be customized thus making fingerprinting difficult
 - You can assign a custom name, IP address or banner
 - You can choose how the system will behave – open, secure, aggressive, failing or strange

9

Honeyd

- Low-interaction so far, open source. Runs on Unix.
- Emulates 17 services, but detects any TCP activity
- Emulates non-existing machines, over 60,000 at a time
 - It can receive traffic through *blackholing* – monitoring non-existing networks for incoming activity
 - Or it can use ARP spoofing to receive traffic only for non-existing machines in a populated network
- It can emulate 473 different operating systems at the same time and at application and TCP stack level
- Only logs transaction data – who attempted the connection and when
- Check it out: <http://www.citj.umich.edu/u/provos/honeyd/>

ManTrap

- High-interaction, commercial. Works on some Solaris.
- Creates up to four instances of *OS cages* on the same machine
 - Have a full functionality of OS but the attacker is unable to go out and compromise other machines
 - Each OS cage can be customized as if they were OS instances on different physical machines
- Runs real applications and also runs real RPC services
- Detects attacks also against closed ports by monitoring network activity
- It can be used to test security solutions
- Can be misused!

11

Honeynets

- High-interaction
- Dedicated machines, with real software, behind a firewall that should control the outbound access
- They are highly flexible
 - Can run any application or OS
 - Can be used as production or research honeypots
- Provide information sharing among security researchers
- Can be used to test new applications
- Highly risky but well controlled and monitored
- High maintenance
- Check it out: <http://project.honeynet.org/>

12

Sweetening the Honeynet

- To make it more alive you can add some features of real-systems
 - Register a domain name
 - Create a believable Web page
 - Create a few users, make them access the machine, create their E-mail folders and make them send/receive E-mails

Value of Honeypots

- They capture a little data but all of it is valuable
 - For detecting new attacks
 - For understanding how attackers work
- They can spend a lot of resources for analysis – no production traffic to log
- No false positives
- They show the size of the threat, unlike firewalls
- Drawbacks:
 - Limited view of things
 - Fingerprinting
 - Risk