Honey POT
Firewall Identification: Banner Grabbing

- Banners are messages sent out by network services during the connection to the service.
- Banners announce which service is running on the system.
- Banner grabbing is a technique generally used by the attacker for OS detection.
- The attacker uses banner grabbing to discover services run by firewalls.
- The three main services that send out banners are FTP, Telnet, and web servers.
- Ports of services such as FTP, Telnet, and web servers should not be kept open, as they are vulnerable to banner grabbing.
- A firewall does not block banner grabbing because the connection between the attacker's system and the target system looks legitimate.
An example of SMTP banner grabbing is: `telnet mail.targetcompany.org 25`. The syntax is:

"< service name > < service running > < port number>"

Banner grabbing is a mechanism that is tried and true for specifying banners and application information.

For example, when the user opens a telnet connection to a known port on the target server and presses Enter a few times, if required, the following result is displayed:

```
C:\>telnet www.corleone.com 80
HTTP/1.0 400 Bad Request
Server: Netscape - Commerce/1.12
```
This system works with many other common applications that respond on a set port.

The information generated through banner grabbing can enhance the attacker's efforts to further compromise the system.

With information about the version and the vendor of the web server, the attacker can further concentrate on employing platform-specific exploit techniques.
Honeypot

A honeypot is an information system resource that is expressly **set up to attract and trap people** who attempt to penetrate an organization’s network.

It has no authorized activity, does not have any production value, and any traffic to it is likely a probe, attack, or compromise.

A honeypot can **log port access attempts, or monitor an attacker’s keystrokes**. These could be early warnings of a more concerted attack.
HoneyPOT

- A honeypot is a system that is intended to attract and trap people who try unauthorized or illicit utilization of the host system.
- Whenever there is any interaction with a honeypot, it is most likely to be a malicious activity.
- Honeypots are unique; they do not solve a specific problem. Instead, they are a highly flexible tool with many different security applications.
- Some honeypots can be used to help prevent attacks; others can be used to detect attacks;
- while a few honeypots can be used for information gathering and research.
Examples:

- Installing a system on the network with no particular purpose other than to log all attempted access.
- Installing an older unpatched operating system on a network.
- For example, the default installation of WinNT 4 with IIS 4 can be hacked using several different techniques.
- A standard intrusion detection system can then be used to log hacks directed against the system and further track what the intruder attempts to do with the system once it is compromised.
- Install special software designed for this purpose. It has the advantage of making it look like the intruder is successful without really allowing him/her access to the network.
Any existing system can be "honeypot-ized."

For example, on WinNT, it is possible to rename the default administrator account and then create a dummy account called "administrator" with no password.

WinNT allows extensive logging of a person's activities, so this honeypot tracks users who are attempting to gain administrator access and exploit that access.
Types of Honeypots

Low-interaction Honeypots
- These honeypots simulate only a limited number of services and applications of a target system or network
- Can not be compromised completely
- Generally, set to collect higher level information about attack vectors such as network probes and worm activities
  - Ex: Specter, Honeyd, and

High-interaction Honeypots
- These honeypots simulate all services and applications
- Can be completely compromised by attackers to get full access to the system in a controlled area
- Capture complete information about an attack vector such as attack techniques, tools and intent of the attack
  - Ex: Symantec Decoy Server and Honeynets
Types of Honeypots

- Honeypots are mainly divided into two types:
- **Low-interaction Honeypot**
  - They work by emulating services and programs that would be found on an individual's system.
  - If the attacker does something that the emulation does not expect, the honeypot will simply generate an error.
- They capture limited amounts of information, mainly transactional data and some limited interaction
- Ex: Specter, Honeyd, and KFSensor
Honeyd

- Honeyd is a **low-interaction honeypot**.
- It is open source and designed to run primarily on UNIX systems.
- Honeyd works on the concept of monitoring **unused IP space**.
- Anytime it sees a connection attempt to an unused IP, it intercepts the connection and then interacts with the attacker, pretending to be the victim.
- By default, Honeyd detects and logs connections to any **UDP or TCP** port.
- In addition, the user can configure emulated services to monitor specific ports, such as an emulated FTP server monitoring **port 21** (TCP).
- When an attacker connects to the emulated service, not only does the honeypot detect and log the activity, but also it captures all of the attacker’s interaction with the emulated service.
In the case of the emulated **FTP server**, an attacker's login and password can be potentially captured; the commands that were issued, what they were looking for, or their identity can be tracked.

Most emulated services work the same way. They expect a specific type of behavior, and then are programmed to react in a predetermined way.
Honeynets are a prime example of a high-interaction honeypot.

A honeynet is neither a product nor a software solution that the user installs.

Instead, it is architecture, an entire network of computers designed to attack.

The idea is to have an architecture that creates a highly controlled network, one where all activity is controlled and captured.

Within this network, intended victims are placed and the network has real computers running real applications.
The "bad guys" find, attack, and break into these systems on their own initiative.

When they do, they do not realize they are within a honeynet.

All of their activity, from encrypted SSH sessions to email and file uploads, is captured without them knowing it by inserting kernel modules on the victim's systems, capturing all of the attacker's actions.

At the same time, the honeynet controls the attacker's activity. Honeynets do this by using a honeywall gateway.

This gateway allows inbound traffic to the victim's systems, but controls the outbound traffic using intrusion prevention technologies.

This gives the attacker the flexibility to interact with the victim's systems, but prevents the attacker from harming other non-honeynet computers.