Administrativia

Mark Allman
mark.allman@case.edu

EECS 325/425
Nov 7 2018

Docket

	This Week	Next Week	Next Week++
Mon	Lecture: Network Layer, DNS	Guest Speaker (Joe Ishac, NASA GRC)	Lecture: CC, CDN, …?
Tue			
Wed	**Lecture: DNS Complexity & Security**	Lecture: DNS Project #4 Due Project #5 Assigned	Lecture: CC, CDN, …?
Thu			Thanksgiving
Fri			

Project 4

- New sample trace: desperado.trace
  - 1.5M packets
- New sample output for thunder.trace

Project 4: Start Slow

Project 4: Start Slow

```
head -1 alive-1.out
1103112609.132870 54 1500 20 T 20 1460
```

Project 4: Start Slow

```
head -1 alive-l.out
1103112609.132870 54 1500 20 T 20 1460

./proj4 -l -t alive.trace |head -1
1103112609.132870 54
```

Project 4: Start Slow

```
head -1 alive-l.out
1103112609.132870 54 1500 20 T 20 1460

./proj4 -l -t alive.trace |head -1
1103112609.132870 54

./proj4 -l -t alive.trace |head -1
1103112609.132870 54 1500
```

Project 4: Start Slow

```
head -1 alive-l.out
1103112609.132870 54 1500 20 T 20 1460

./proj4 -l -t alive.trace |head -1
1103112609.132870 54

./proj4 -l -t alive.trace |head -1
1103112609.132870 54 1500

awk ‘{print $1,$2,$3}’ alive-l.out |head -1
```

Project 4: Start Slow

```
head -1 alive-l.out
1103112609.132870 54 1500 20 T 20 1460

./proj4 -l -t alive.trace |head -1
1103112609.132870 54

./proj4 -l -t alive.trace |head -1
1103112609.132870 54 1500

awk ‘{print $1,$2,$3}’ alive-l.out |head -1
1103112609.132870 54 1500
```

Project 4: Start Slow

```
head -1 alive-l.out
1103112609.132870 54 1500 20 T 20 1460

./proj4 -l -t alive.trace |head -1
1103112609.132870 54

./proj4 -l -t alive.trace |head -1
1103112609.132870 54 1500

awk ‘{print $1,$2,$3}’ alive-l.out |head -1
1103112609.132870 54 1500
awk ‘{print $1,$2,$3}’ alive-l.out > expected.out
```

Project 4: Start Slow

```
head -1 alive-l.out
1103112609.132870 54 1500 20 T 20 1460

./proj4 -l -t alive.trace |head -1
1103112609.132870 54

./proj4 -l -t alive.trace |head -1
1103112609.132870 54 1500

awk ‘{print $1,$2,$3}’ alive-l.out |head -1
1103112609.132870 54 1500
awk ‘{print $1,$2,$3}’ alive-l.out > expected.out
./proj4 -l -t alive.trace > my-l.out
```

Project 4: Start Slow

```
head -1 alive-l.out
1103112609.132870 54 1500 20 T 20 1460

./proj4 -l -t alive.trace |head -1
1103112609.132870 54

./proj4 -l -t alive.trace |head -1
1103112609.132870 54 1500

awk ‘{print $1,$2,$3}’ alive-l.out |head -1
1103112609.132870 54 1500
awk ‘{print $1,$2,$3}’ alive-l.out > expected.out
./proj4 -l -t alive.trace > my-l.out

diff expected.out my-l.out
```

Project 4: Sanity Checking

Project 4: Sanity Checking

- Leverage the "overlap" between modes

Project 4: Sanity Checking

- Leverage the "overlap" between modes

- E.g., -s and -l will both tell you how many packets there are in the file

Project 4: Sanity Checking

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3
```

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3

./proj4 -t shout.trace -l |wc -l
```

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3

./proj4 -t shout.trace -l |wc -l
4
```

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3

./proj4 -t shout.trace -l |wc -l
4
```

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3

./proj4 -t shout.trace -l |wc -l
4

./proj4 -t shout.trace -l |head -1
```

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3

./proj4 -t shout.trace -l |wc -l
4

./proj4 -t shout.trace -l |head -1
1103112609.132870 54 1500 20 T 20 1460
```

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3
```

```
./proj4 -t shout.trace -l |wc -l
4
```

```
./proj4 -t shout.trace -l |head -1
1103112609.132870 54 1500 20 T 20 1460
```

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3

./proj4 -t shout.trace -l |wc -l
4

./proj4 -t shout.trace -l |head -1
1103112609.132870 54 1500 20 T 20 1460
./proj4 -t shout.trace -l |tail -1
```

Allman

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3

./proj4 -t shout.trace -l |wc -l
4

./proj4 -t shout.trace -l |head -1
1103112609.132870 54 1500 20 T 20 1460
./proj4 -t shout.trace -l |tail -1
1103112611.247566 42 56 20 ? ? ?
```

Project 4: Sanity Checking

```
./proj4 -t shout.trace -s
TIME SPAN: 1103112609.132870 - 1103112611.247566
TOTAL PACKETS: 4
IP PACKETS: 3

./proj4 -t shout.trace -l |wc -l
4

./proj4 -t shout.trace -l |head -1
1103112609.132870 54 1500 20 T 20 1460
./proj4 -t shout.trace -l |tail -1
1103112611.247566 42 56 20 ? ? ?
```

Project 4: Sanity Checking

Project 4: Sanity Checking

```
./proj4 -t kashmir.trace -p |head -1
```

Project 4: Sanity Checking

```
./proj4 -t kashmir.trace -p |head -1
1103112609.132870 10.1.124.10 575 192.168.2.16 …
```

Project 4: Sanity Checking

```
./proj4 -t kashmir.trace -p |head -1
1103112609.132870 10.1.124.10 575 192.168.2.16 …
./proj4 -t kashmir.trace -p |awk '{print $2,$4}' \
   |sort -u > pairs1.out
```

Project 4: Sanity Checking

```
./proj4 -t kashmir.trace -p |head -1
1103112609.132870 10.1.124.10 575 192.168.2.16 …
./proj4 -t kashmir.trace -p |awk '{print $2,$4}' \
  |sort -u > pairs1.out

./proj4 -t kashmir.trace -m |head -1
```

Project 4: Sanity Checking

```
./proj4 -t kashmir.trace -p |head -1
1103112609.132870 10.1.124.10 575 192.168.2.16 …
./proj4 -t kashmir.trace -p |awk '{print $2,$4}' \
  |sort -u > pairs1.out

./proj4 -t kashmir.trace -m |head -1
10.1.124.10 192.168.2.16 2815696
```

Project 4: Sanity Checking

```
./proj4 -t kashmir.trace -p |head -1
1103112609.132870 10.1.124.10 575 192.168.2.16 …
./proj4 -t kashmir.trace -p |awk '{print $2,$4}' \
  |sort -u > pairs1.out

./proj4 -t kashmir.trace -m |head -1
10.1.124.10 192.168.2.16 2815696
./proj4 -t kashmir.trace -m |awk '{print $1,$2}' \
  |sort > pairs2.out
```

Project 4: Sanity Checking

```
./proj4 -t kashmir.trace -p |head -1
1103112609.132870 10.1.124.10 575 192.168.2.16 …
./proj4 -t kashmir.trace -p |awk '{print $2,$4}' \
  |sort -u > pairs1.out

./proj4 -t kashmir.trace -m |head -1
10.1.124.10 192.168.2.16 2815696
./proj4 -t kashmir.trace -m |awk '{print $1,$2}' \
  |sort > pairs2.out

diff pairs1.out pairs2.out
```

Project 4: Altering Input

Project 4: Altering Input

- Traffic matrix mode requires state

Project 4: Altering Input

- Traffic matrix mode requires state

- Even large packet traces can have modest state requirements
  - e.g., desperado.trace contains 1.5M packets, but only ≈500 uni-directional flows

Project 4: Altering Input

- Traffic matrix mode requires state
- Even large packet traces can have modest state requirements
  - e.g., desperado.trace contains 1.5M packets, but only ≈500 uni-directional flows
- Can simulate larger state requirements
  - after reading a packet, *change it!*

Project 4: Altering Input

```
while (next_packet (fd,&pinfo))
{




    // do something
}
```

Project 4: Altering Input

```
unsigned long next_fake_ip = 0;

while (next_packet (fd,&pinfo))
{





    // do something
}
```

Project 4: Altering Input

```
unsigned long next_fake_ip = 0;

while (next_packet (fd,&pinfo))
{


        pinfo.iph->saddr = next_fake_ip++;
        pinfo.iph->daddr = next_fake_ip++;

    // do something
}
```

Project 4: Altering Input

```
unsigned long next_fake_ip = 0;

while (next_packet (fd,&pinfo))
{
    if (pinfo.iph != NULL)
    {
        pinfo.iph->saddr = next_fake_ip++;
        pinfo.iph->daddr = next_fake_ip++;
    }
    // do something
}
```

Project 4: Altering State

Project 4: Altering State

- desperado.trace contains 1.5M packets, but only ≈500 uni-directional flows

Project 4: Altering State

- desperado.trace contains 1.5M packets, but only ≈500 uni-directional flows
- Using this strategy we can have 1.5M uni-directional flows for testing

Project 4: Altering State

- desperado.trace contains 1.5M packets, but only ≈500 uni-directional flows
- Using this strategy we can have 1.5M uni-directional flows for testing
- Increase state requirements by 4 orders of magnitude
  - good test!

Project 4

- -s: summary mode
- -l: length mode
- -p: packet printing mode
- -m: traffic matrix mode

Project 4

- ✓ -s: summary mode
- ✓ -l: length mode
- ✓ -p: packet printing mode
- ✓ -m: traffic matrix mode

Questions ??