SAN JOSÉ STATE UNIVERSITY

First Experiences with DETER Testbed

Dr. Xiao Su
Computer Engineering
San Jose State University
http://www.sjsu.edu/people/xiao.su

NSF TRUST WISE 2012

Agenda

SAN JOSÉ STATE UNIVERSITY

- **Prior experience with PlanetLab**
  - Setup
  - Challenges in using PlanetLab in security research/education
- **DETER Testbed**
  - Overview
  - Setting up DETER experiments
  - Running DETER experiments
- **Demo**

Our Network Security Class

SAN JOSÉ STATE UNIVERSITY

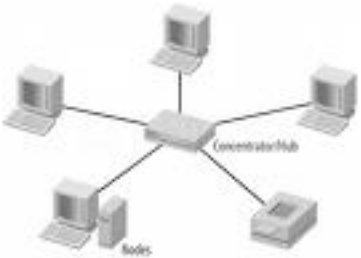
- **How attackers think and work?**
  - Attack phases
- **What are the tools and algorithms to counter such attacks?**
  - Crypto tools
  - Authentication
  - Access control
  - Key distribution

Experimental Setups

SAN JOSÉ STATE UNIVERSITY

- **Personal computer**

- **Local testbed**
  - Isolated from campus networks



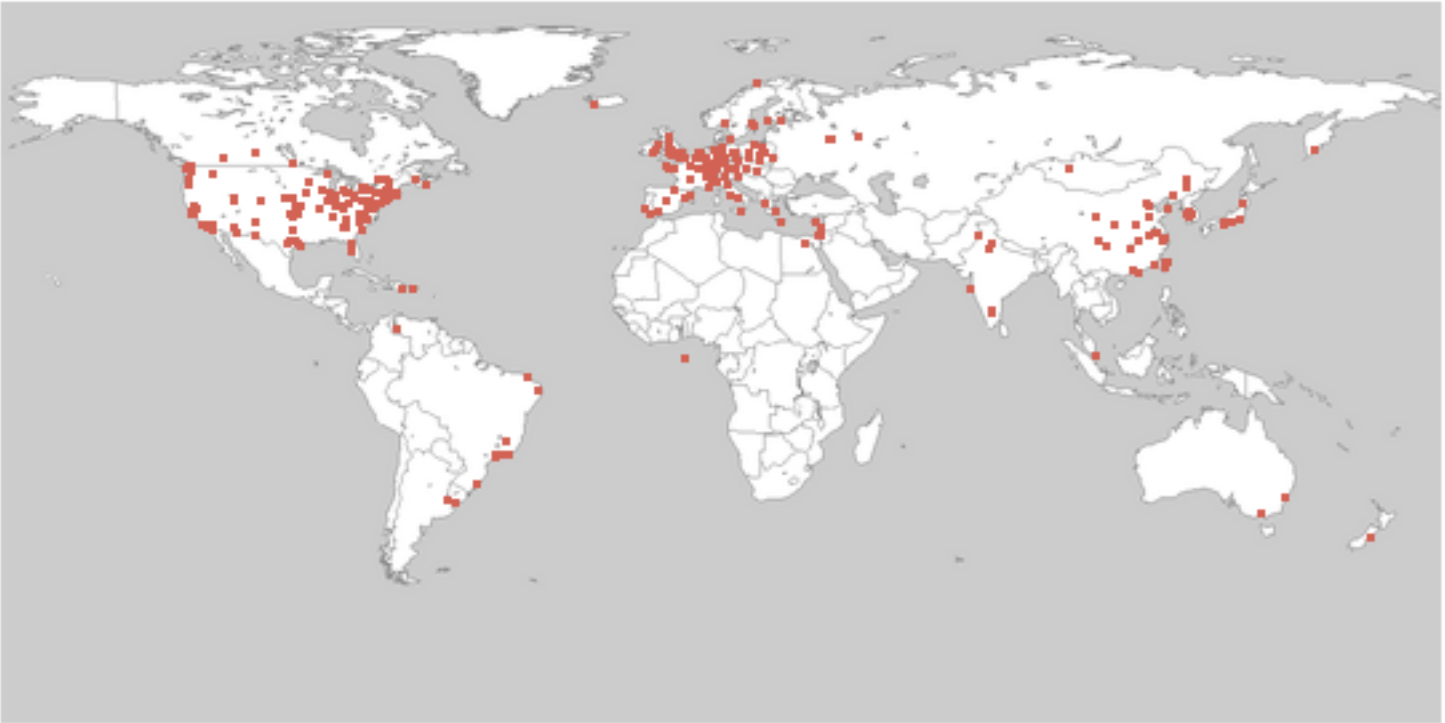
- **Internet scale testbed**

Challenges in Teaching Network Attacks

- **Many tools need admin privilege**
  - Sniffing, spoofing, MIM,…
- **Attacking experiments hard to set up**
  - Not interfering with normal network operations
  - Contained in a closed network
  - Some experiments heavily depend on compiler, OS types (e.g., buffer overflow)
- **Hard to conduct security experiments in the Internet scale?**
  - Evaluate how effective anti-DDoS schemes are?
- **Need to use Internet scale testbed!!**

Internet Scale Testbed - PlanetLab

SAN JOSÉ STATE UNIVERSITY



Picture taken from http://www.planet-lab.org

- **1127 nodes at 544 sites (as of June 18, 2012)**

Running Services in PlanetLab

SAN JOSÉ STATE UNIVERSITY

- **First, create a slice (by PI only)**
  - UNIX shell access to a set of PlanetLab nodes
  - virtual machines
  - Faults or misbehaviors can be isolated and traced back to virtual machines
- **Create public/private key pair (by users)**
- **Upload public key to PlanetLab (by users)**
- **Remote access machines using ssh with slice name as the user name**
- **Install packages, deploy and run services**

Why Not PlanetLab in Security Experiments

SAN JOSÉ STATE UNIVERSITY

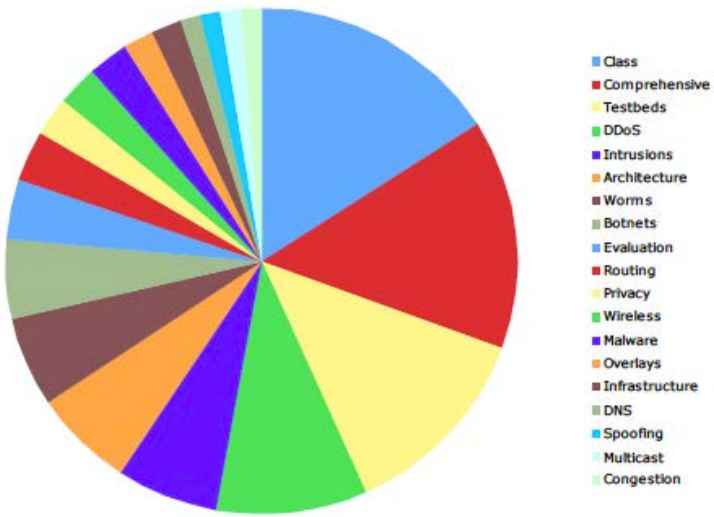
- **PlanetLab nodes are part of the Internet**
  - Traffic impact the Internet: what happens when you experiment DoS on PlanetLab?
  - The testbed is closely monitored by a team of professionals, for network attacks, worm propagation, copyright infringement, and other malicious traffic.
- **PlanetLab nodes run uniform Linux-based OS**
  - Not possible if experiments require different types of OS
  - OS and network configurations on PlanetLab nodes are not customizable
- **Operational concerns**
  - Nodes in the same experiment (slices) don't share home directories: lots of ssh copies
  - A good percentage of nodes not accessible by ssh, due to different ssh policies and versions

Here We Have DETER

SAN JOSÉ STATE UNIVERSITY

Where is DETER Used?

Types of DETER Projects



- **DETER is a public testbed to run repeatable security experiments**
  - Based on Univ. of Utah's Emulab
  - Specially enhanced for security research/education
  - Jointly run by USC's ISI and UC Berkeley

PlanetLab vs. DETER

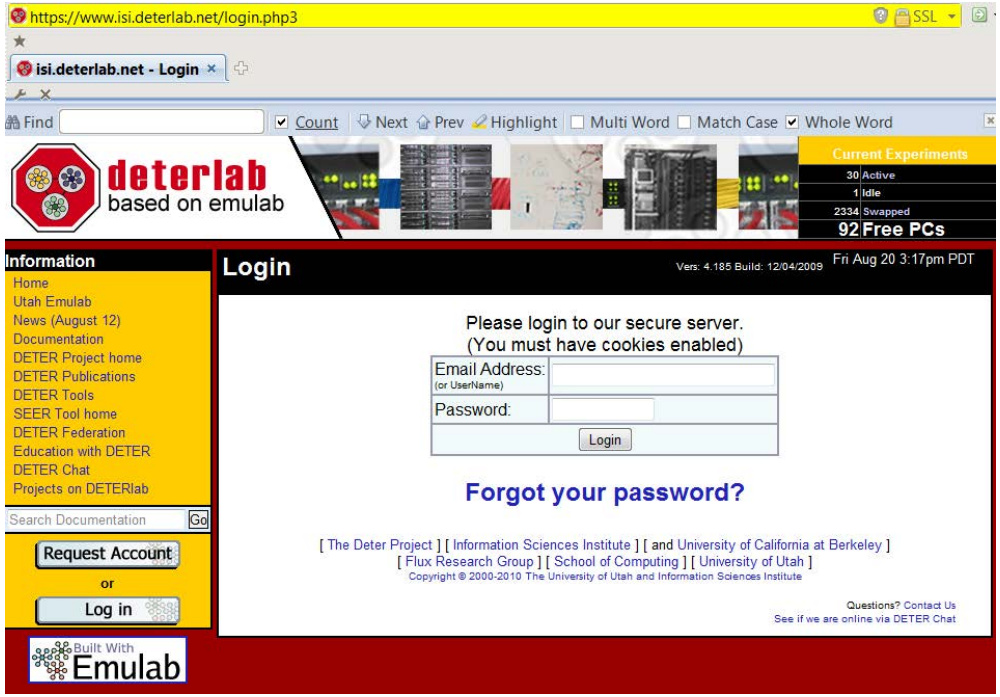
SAN JOSÉ STATE UNIVERSITY

	PlanetLab	DETER
Isolated or Open Network	Part of the Open Internet	Isolated Testbed
OS Uniformity	Only runs Linux OS (FC based)	Supports a large list of OS types
OS Customization	Not possible	Supports customized OS images
Network Configuration	Static, part of the Internet	Supports configuration of network, defined by NS scripts
Home directory	Different nodes have different home directories	Same home directory for all nodes
User Management	Site PI's task	DETER support

Accessing DETER Testbed

SAN JOSÉ STATE UNIVERSITY

- **Login to https://www.isi.deterlab.net/login.php3**



DETER Experiments

SAN JOSÉ STATE UNIVERSITY

lab | Logout | News NEW! | Contact Us Search Documentation Go

n Experimentation

166 Free PCs
1 PCs reloading
40 active users
35 active expts.

News:

DETER Chat (IRC) no longer official support medium (2011-04-14)
Web Login by email address is no longer supported. (2011-03-23)
CSET'11 Call for Papers (2011-02-07)

Full news stories

Scheduled downtime: Wednesdays: 5PM-7PM, Saturdays: 10AM-1PM Pacific Time.

Experiments | Projects | Profile

Current Experiments

PID	EID	State	Nodes [1]	Hours Idle [2]	Description
SJSUCMPE209	BufferOverflow1004	swapped	1		Single node buffer overflow experiments on Ubuntu 10.04
SJSUCMPE209	SampleExp	swapped	3		To experiment with buffer overflow on Ubuntu 8.04
SJSUCMPE209	switchf2009	swapped	4		Experimenting with Active Sniffing
SJSUCMPE209	synflood	swapped	5		TCP SYN flooding attack

1. Node counts in **green** show a rough estimate of the minimum number of nodes required to swap in. They account for delay nodes, but not for node types, etc.
2. A **?** indicates that the data is stale, and at least one node in the experiment has not reported on its proper schedule.

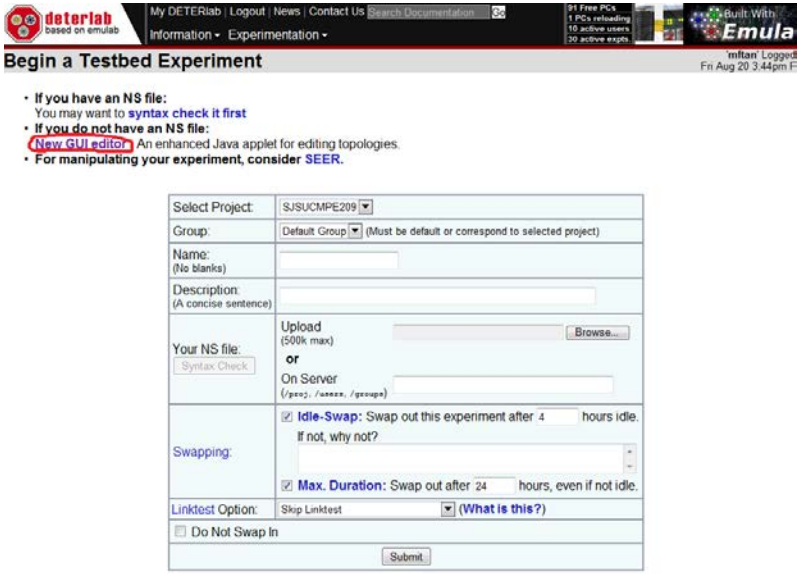
Beginning a New DETER Expeirment

- **Parameters to fill in**
  - Name of your experiment
  - Description of your experiment
  - Your NS file to specify a network topology
    - Don't know NS? Use the GUI editor!

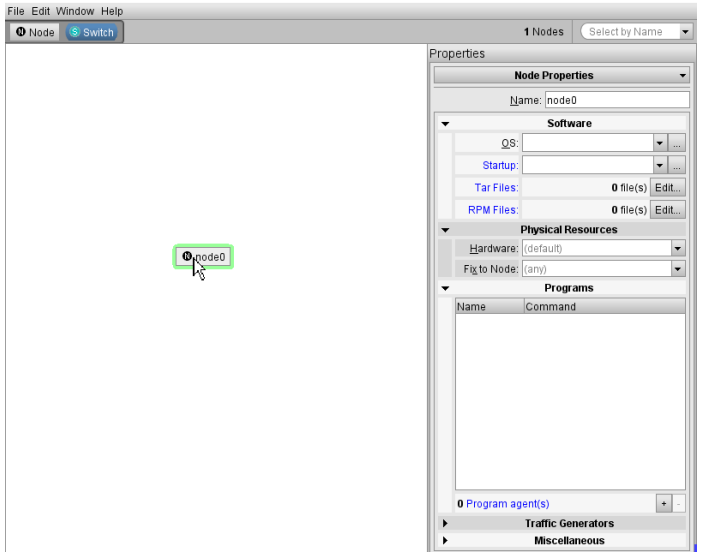
Beginning a New DETER Experiment

SAN JOSÉ STATE UNIVERSITY

- Beginning experiment page

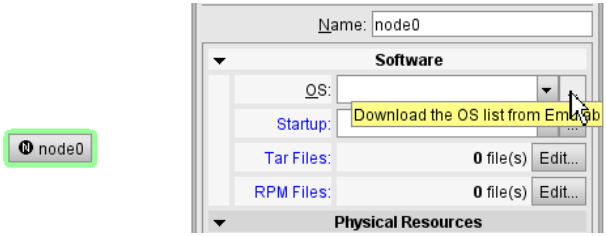


- This is how a GUI editor looks like

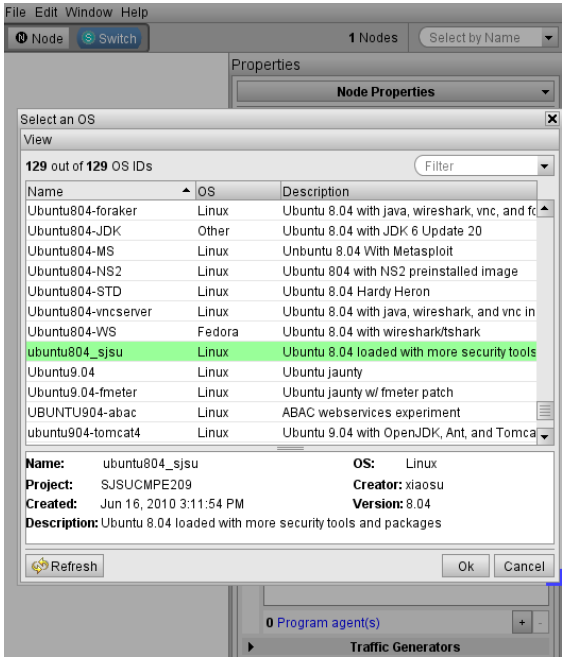


Configuring OS and Network Settings

- Configuring OS

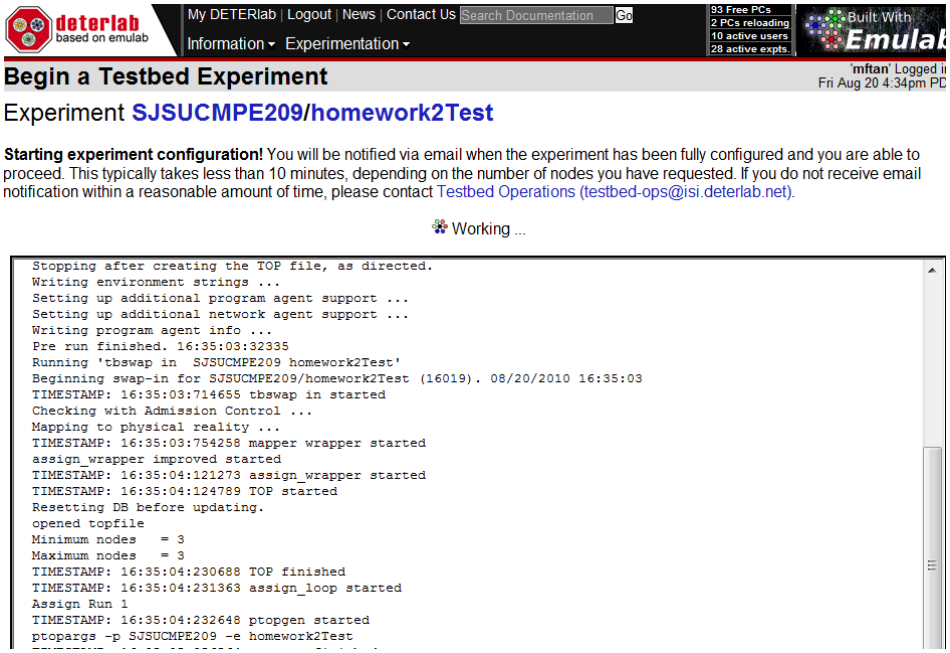


- List of supported OS

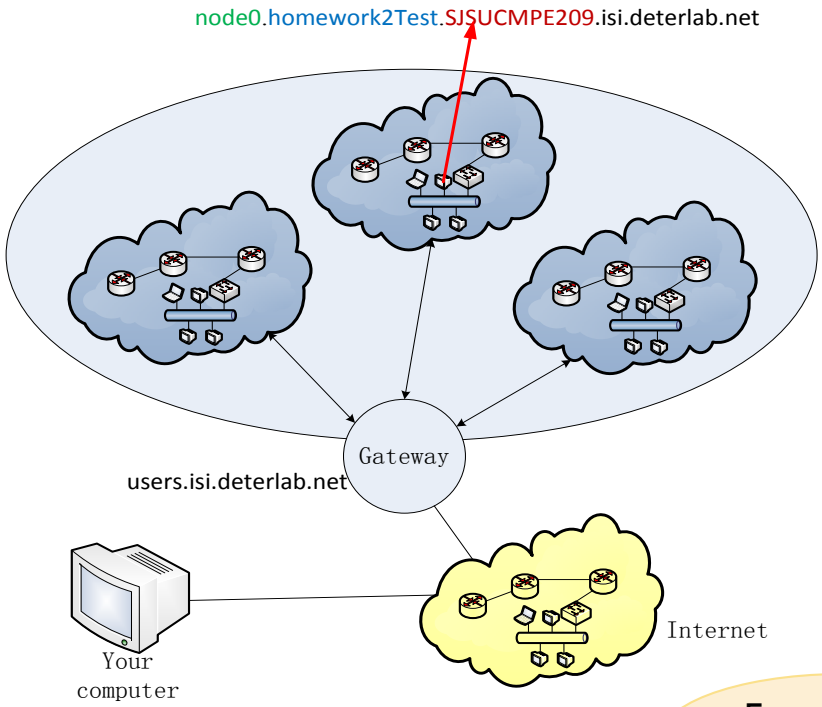


Next Step: Swap Your Experiment In

- **Be patient, may take sometime**



Accessing Nodes in Your Experiment



- **First log into the gateway node**
  - ssh users.isi.deterlab.net
- **On gateway, accessing your nodes by**
  - ssh node0.homework2Test.SUSUCMPE209.isi.deterlab.net

Experiment name

Project name

Cleaning Up

SAN JOSÉ STATE UNIVERSITY

- Swapping out your experiment
- Terminating an experiment

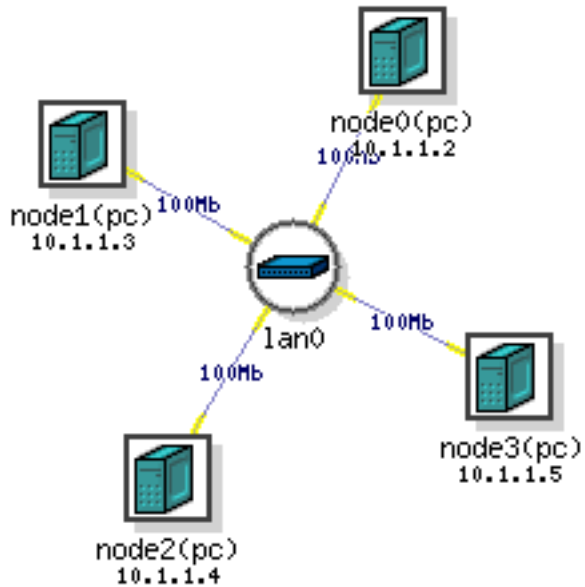
How Did We Use DETER in Our Class

SAN JOSÉ STATE UNIVERSITY

- **We used DETER**
  - In-class demos
  - Assignments
    - Buffer overflow (Ubuntu 8.04LTS, gcc 3.x) – This combination of gcc compiler and OS version is hard to find now
    - TCP SYN flooding – Not possible on regular lab environments
    - Man-in-the-middle attack

Man-in-the-Middle Experiment on DETER

- **Network configuration**
  - 4 nodes interconnected using a switch
  - 3 nodes running Ubuntu 8.04LTS, and 1 node running Ubuntu 10.04



Can Node0 Sniff Other Nodes' Packets?

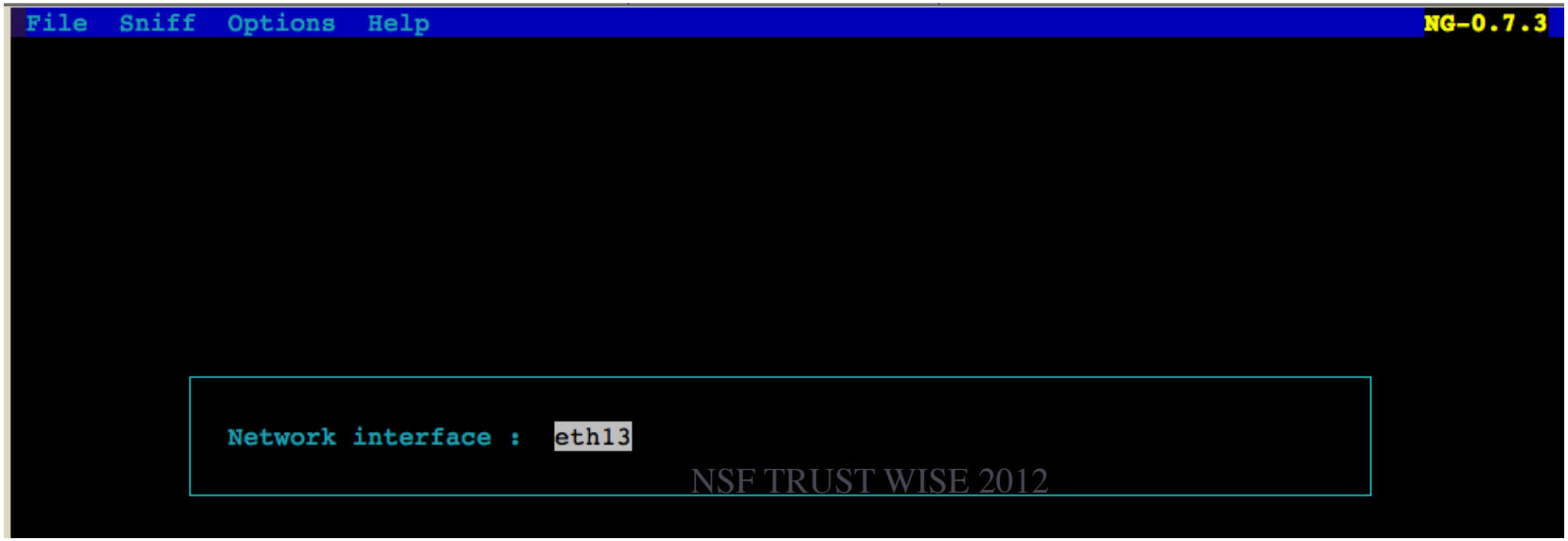
- **Logging into Node0 in the experiment**

```
Macintosh-6:~ xsu$ ssh -l xiaosu users.isi.deterlab.net
Password:
Last login: Fri Apr 29 14:29:50 2011 from x130-65-152-108
> ssh node0.switchf2009.sjsucmpe209
Linux node0.switchf2009.sjsucmpe209.isi.deterlab.net 2.6.24-23-deter #4 SMP
Wed Jan 21 23:15:52 MST 2009 i686
node0:~>
```

Can Node0 Sniff Other Nodes' Packets?

- **Starting ettercap**

Macintosh-6:~ xsu$ **sudo ettercap -C**



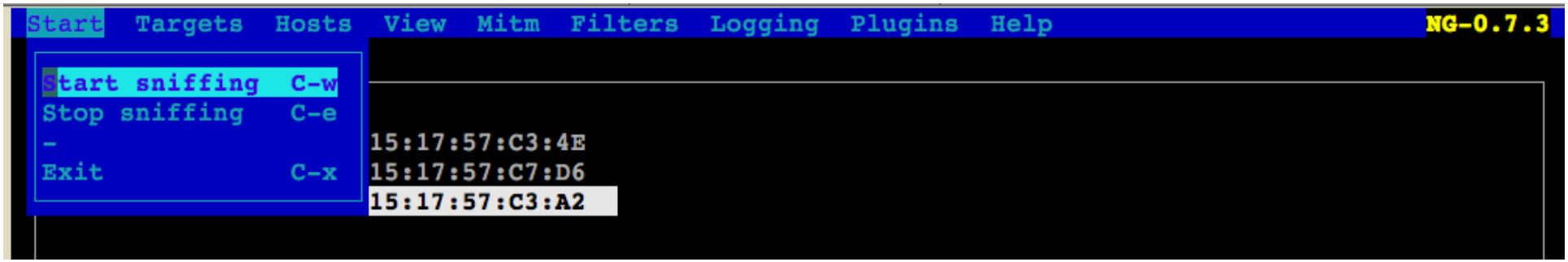
Can Node0 Sniff Other Nodes' Packets?

- Scan for hosts

```
Start  Targets  Hosts  View  Mitm  Filters  Logging  Plugins  Help          NG-0.7.3

 Hosts list...

  10.1.1.3          00:15:17:57:C3:4E
  10.1.1.4          00:15:17:57:C7:D6
  10.1.1.5          00:15:17:57:C3:A2

 User messages:
1698 tcp OS fingerprint
2183 known services
Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
3 hosts added to the hosts list...
```

Can Node0 Sniff Other Nodes' Packets?

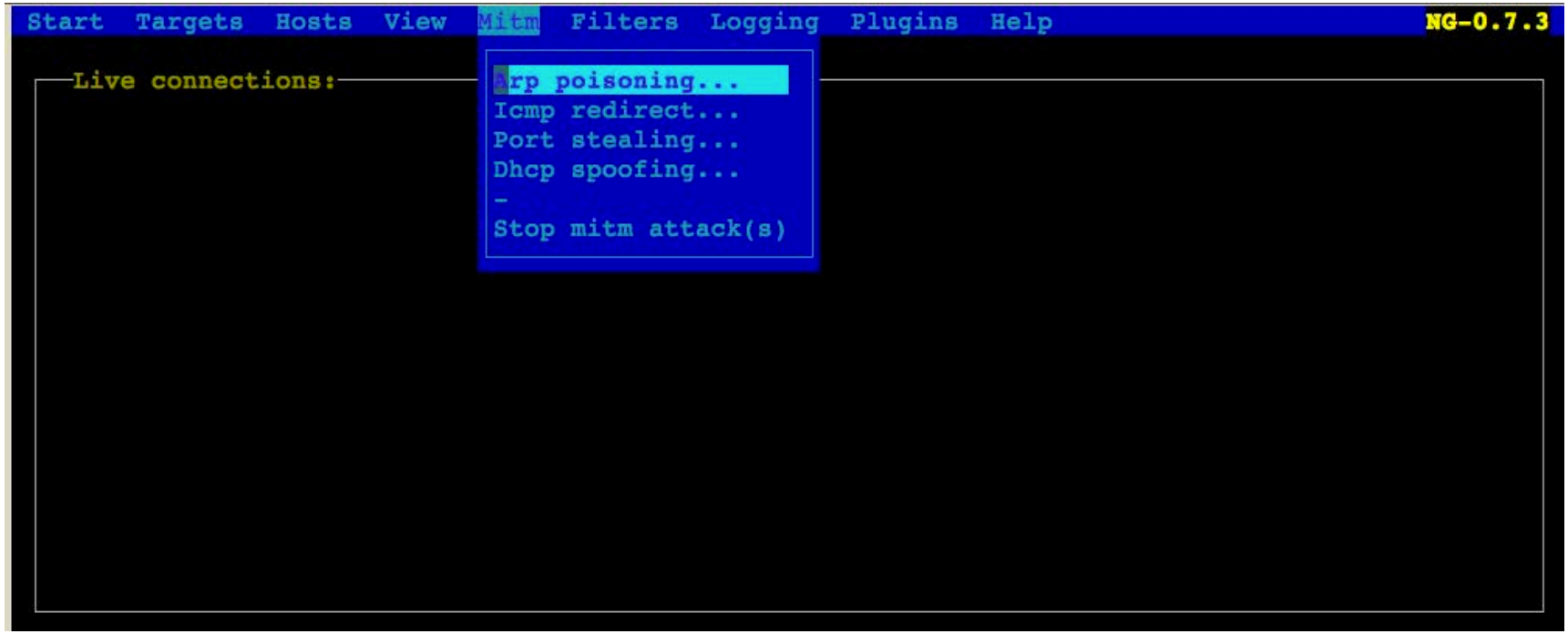
- Start sniffing
- Generate packets from node1 to node2



node1:~> nc node2 54321
1st try: can you see me?

node2:~> nc -l -p 54321
1st try: can you see me?

Now Enabling ARP Poisoning



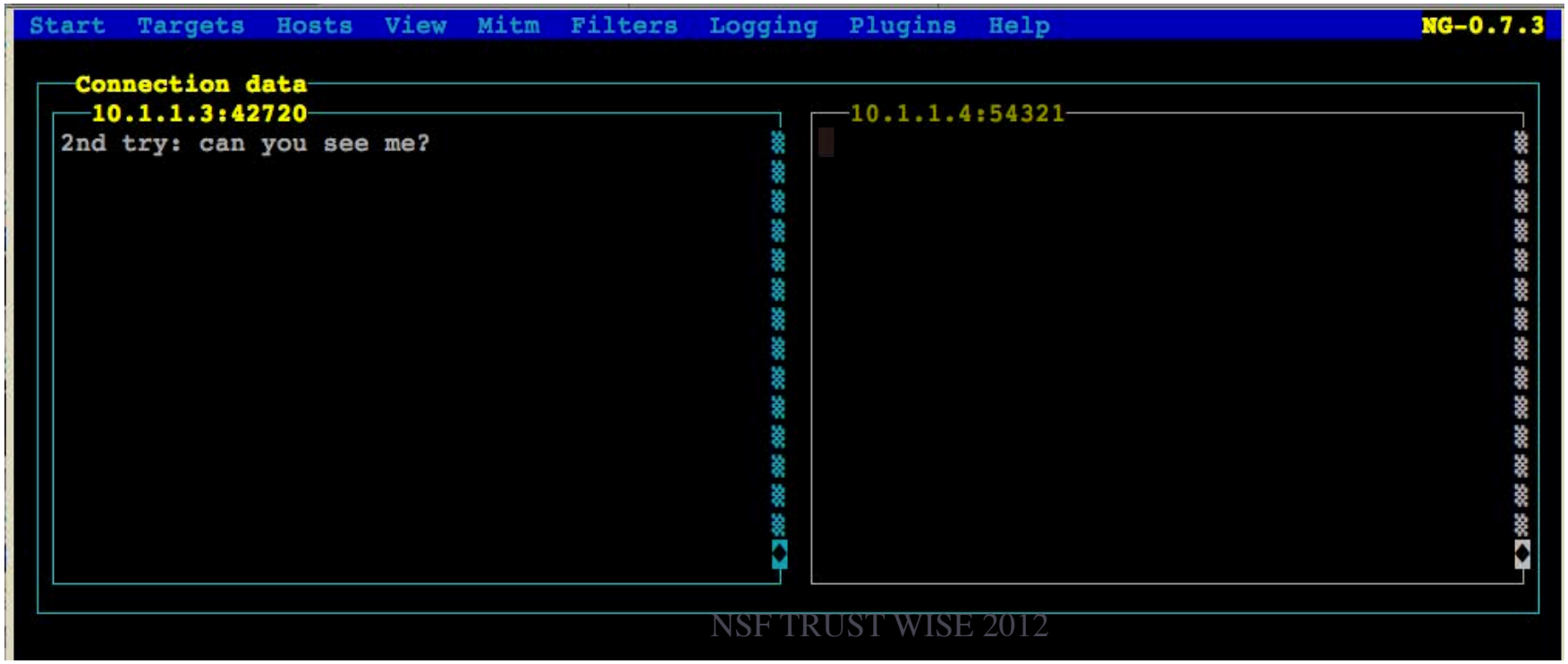
What Will Happen Next?

SAN JOSÉ STATE UNIVERSITY

- Sending something from node1 to node2

node1:~> nc node2 54321
1st try: can you see me?
2nd try: can you see me?

node2:~> nc -l -p 54321
1st try: can you see me?
2nd try: can you see me?



Modifying Packet

SAN JOSÉ STATE UNIVERSITY

- **Step 1: defining a filter**
  - What does the filter do?

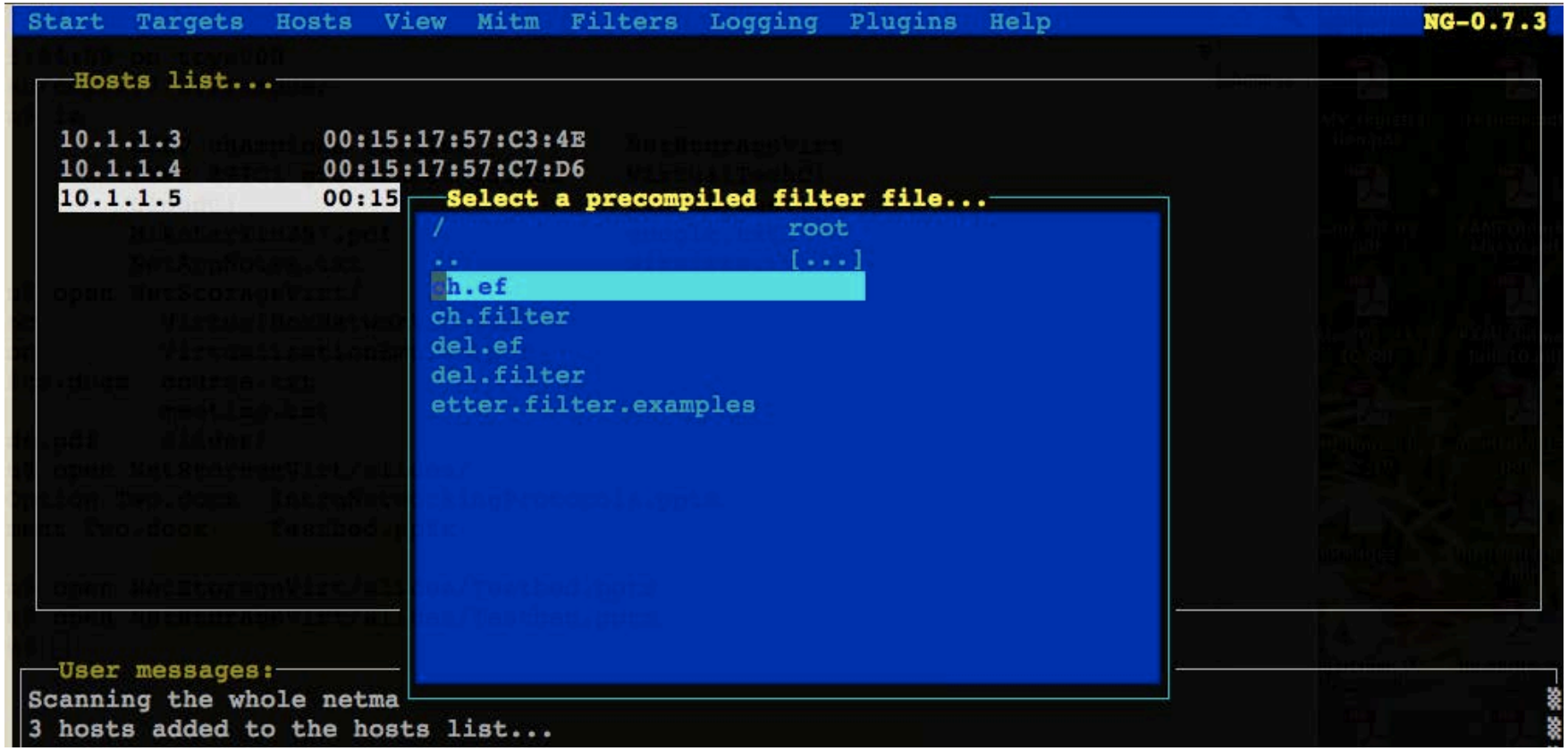
```
node2:~> cat ASniffing/ch.filter
# change TCP payload
if (tcp.dst == 54321 && search(DATA.data, "install")) {
   replace("install", "upgrade");
}
```

- **Step 2: compiling the filter**

```
node2:~/ASniffing> etterfilter -o ch.ef ch.filter
```

Modifying the Packet

- **Step 3: load the filter**



Will Packets Be Modified?

SAN JOSÉ STATE UNIVERSITY

- Packets sent and received, after loading the filter

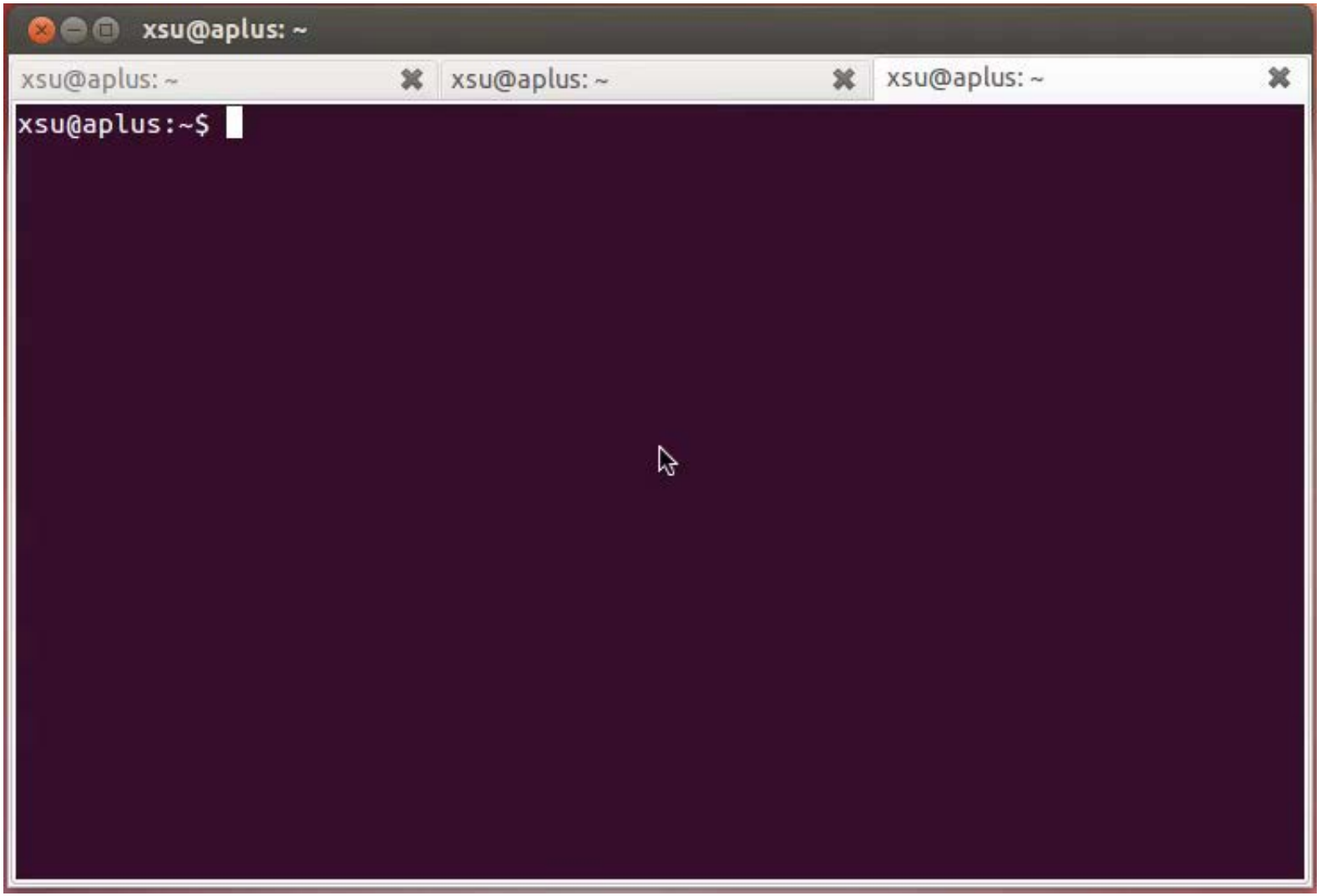
```
node1:~> nc node2 54321
1st try: can you see me?
2nd try: can you see me?
apt-get install gcc4
```

```
node2:~> nc -l -p 54321
1st try: can you see me?
2nd try: can you see me?
apt-get upgrade gcc4
```

DEMO Video

SAN JOSÉ STATE UNIVERSITY

- **Video capturing the above experiments**
  - Sniffing without arp spoofing
  - Sniffing with arp spoofing enabled
  - Sniffing with arp spoofing enabled and filter loaded to modify packet


xsu@aplus: ~
xsu@aplus: ~
xsu@aplus: ~
xsu@aplus: ~
xsu@aplus:~$

SAN JOSÉ STATE UNIVERSITY

Questions?