# Attack Class: Address Spoofing

L. Todd Heberlein

23 Oct 1996

Net Squared Inc.
todd@NetSQ.com

# Overview of Talk

- Introduction
- Background material
- Attack class
- Example attack
- Popular questions
- Extensions

# UCD Vulnerabilities Group

- UCD's vulnerabilities group studies attacks and their underlying vulnerabilities for the purpose of modeling them. We believe a sufficiently complete model will allow us to both predict new instances of general attack classes and build generic schemes for detecting exploitations of general vulnerability classes.

# Address Masquerading

- Many of today's network services use host names or addresses for both identification AND authentication.
- Examples: rlogin, rsh, mountd, wrappers, firewalls
- Higher level services use these lower level services (e.g., backups)

# History of Talk

- R.T. Morris, 85
- S. Bellovin, 89
- UCD Discussed, Feb. 94
- UCD Presented, Mar 94
- Mitnick-Tsutomu, Dec 94
- UCD paper, spring 95
- Mendax, Rbone, summer 95
- Wee (UCD), fall 95
- USAF project, Jan. 96

# Orders and Dialogues

- Need better names
  - » asynchronous vs. synchronous
  - » connectionless vs. connection-oriented
- An order is a request requiring only a single “message”.
- A dialogue is a request which requires the exchange of several, interdependent “messages”.
- From recipient’s point of view

# Connectionless Communication (Orders)

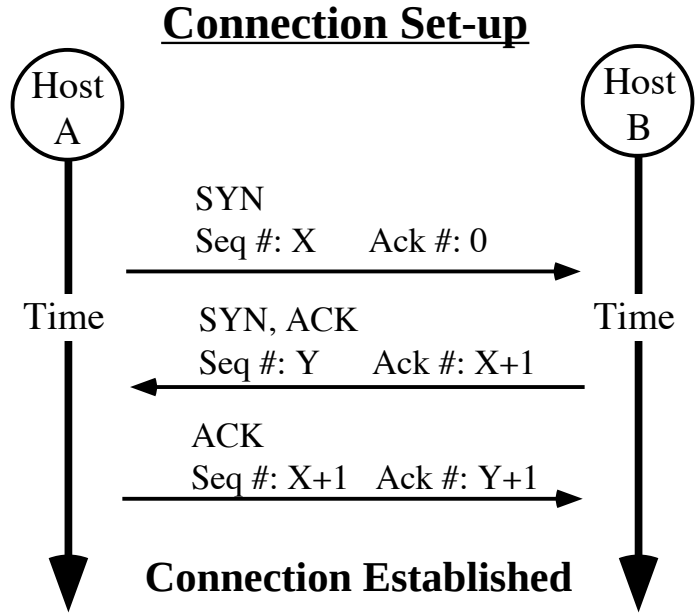
- Connectionless communication (e.g., supplied by UDP), does not keep state information
- No guarantee of delivery or order
- Efficient in many environments
- RPC on UDP (NFS)

# Connection-oriented Communication (Dialogues)

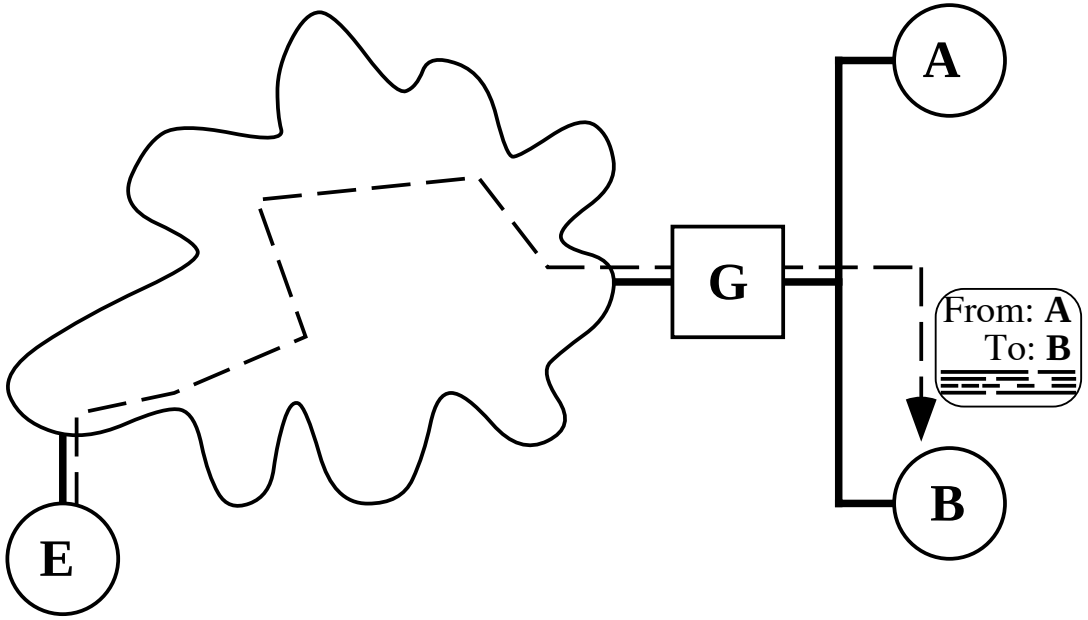
- Additional state information kept, representing a limited history of communication
- Provides “guarantee” that information will both arrive and arrive in order
- May require more resources and be less efficient in some environments

# TCP/IP Example

- Three phases: set-up, data exchange, tear-down
- set-up is a three-way handshake
- Third packet requires information from second packet.



# Routing in an internet




- Host constructs packet and simply places it on the network
- As the packet travels across the internet, only the destination address is used

# The Attack

- Definition of what an attack is
- Restrictions to be concerned with
- Strategy of the attacker

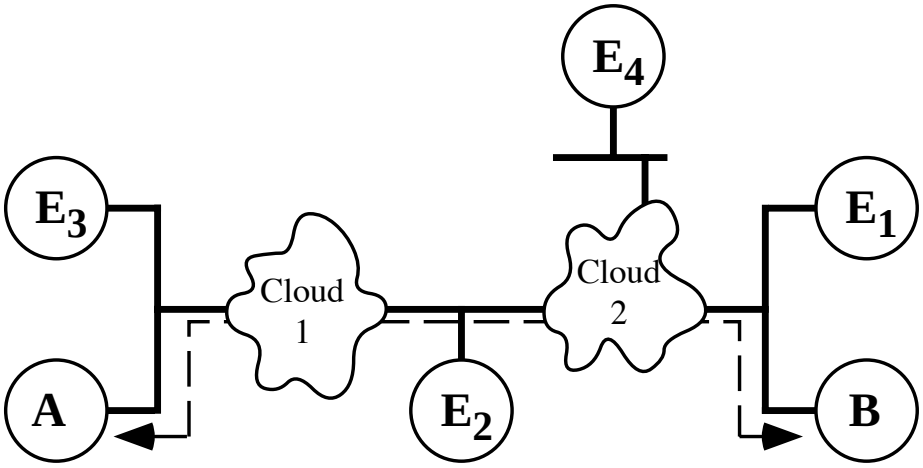
# Definition of Attack

- Players: Alice (A), Bob (B), and Eve (E)
- Bob grants Alice special privileges by listing Alice's address or name in a special file
- Eve is the villain
- Eve's goal: **To get Bob to perform a specific action that he would perform for Alice but not Eve**

# Restrictions

- The placement of Alice, Bob, and Eve (the topology)
- The nature of the communication required by Eve to carry out the attack.
- These restrictions will help define Eve's strategy

# Architecture (or Topology)




- Alice and Bob on separate networks; Eve in one of four locations
- Other architectures are simply special cases of this one

# Nature of Communication

- Eve's communication must be indistinguishable from Alice's communication with Bob
- Order communication
  - » request is carried out immediately
  - » No role-backs
- Dialogue communication
  - » must make sense to Bob
  - » Alice cannot be allowed to interfere

# Eve's Strategy

- Establish a forged communication with Bob
- Prevent Alice from alerting Bob until it is too late

# Establishing a Forged Communication

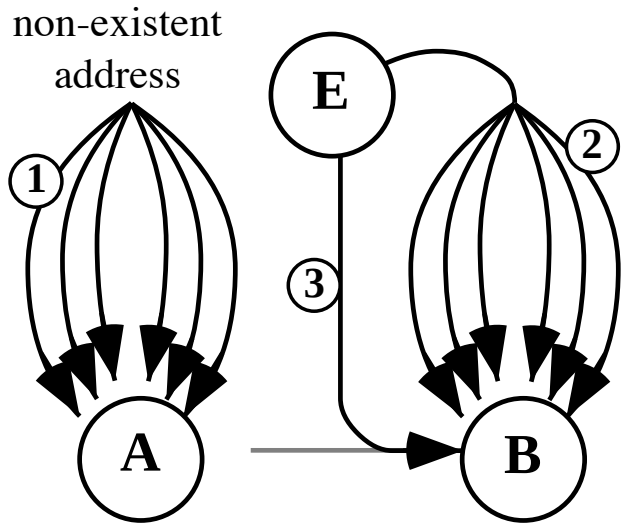
- Construct packet, and place it on the network. The network will deliver it for Eve
- For order-based communication, the communication is done
- For dialogue-based communication, further messages must be exchanged
  - » if Eve is in $E_1$, $E_2$, or $E_3$, further communication is easy
  - » if Eve is in $E_4$, she must either modify the messages' routes, or predict what the messages will contain

# Prevent Alice from Interfering

- Prevent Bob's packets from reaching Alice (or Alice's from reaching Bob)
- Take away Alice's ability to respond
  - » wait for Alice to go down for maintenance
  - » force Alice to crash
  - » block part of Alice's operating system from processing Bob's packets (graceful ??)
- Complete communication before Alice can respond

# Example Attack

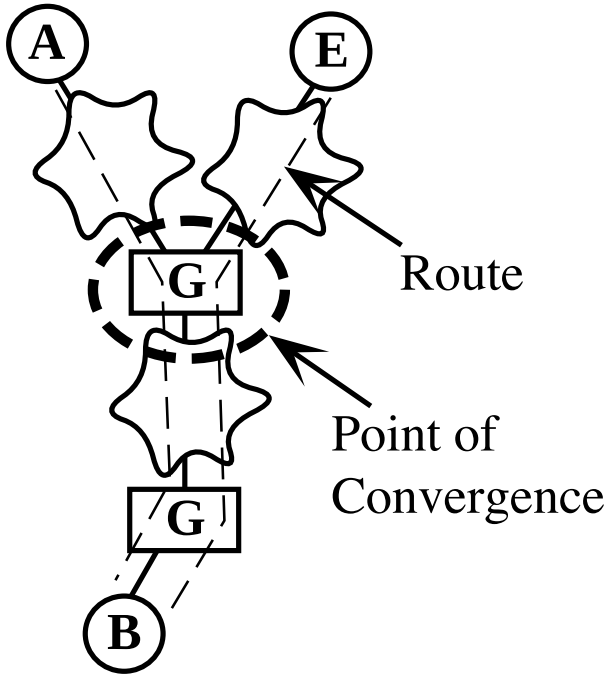
| | | |
|---|---|---|
| **Players** | **E** | adversary |
| | **A** | server |
| | **B** | X-client |
| **Steps** | **1** | Prevent Alice From Responding |
| | **2** | Probe for sequence number prediction |
| | **3** | Forge communication |




- Used against Tsutomu Shimamura, attributed to Kevin Mitnick
- Detailed ten years earlier by R.T. Morris

# Questions

- Couldn't this attack be stopped by simply configuring routers not to forward obviously forged packets?



# Questions cont.

- Couldn't we require all "trusted" hosts to belong to the same physical network and use lower level addresses (e.g., ethernet)?

```
ie(7D)                          Devices                          ie(7D)

NAME
     ie - Intel 82586 Ethernet device driver

SYNOPSIS
     /dev/ie

DESCRIPTION

...
     The DL_SET_PHYS_ADDR_REQ primitive changes the 6 octet  Eth-
     ernet   address  currently  associated  (attached)  to  this
     stream.  The credentials of  the  process  which  originally
```

# Questions cont.

- **Couldn't we simply write a more secure algorithm for choosing initial sequence numbers?**
- Only if Eve is NOT is position $E_1$, $E_2$, or $E_3$, and Eve is NOT able to alter the path of Bob's messages to Alice (e.g., source routing or routing table modification). Also, this solution does not apply to order-based communications.

# Extensions to this Attack: Session Hijacking

- One-time authentication services are vulnerable
- Commercial programs exist which do session hijacking
- Demonstrated against systems with challenge-response authentication

# Extensions cont.

- Eve's goal: **To get Bob to accept information he would only accept from Alice**