802.11 Denial-of-Service Attacks Real Vulnerabilities and Practical Solutions

John Bellardo and Stefan Savage

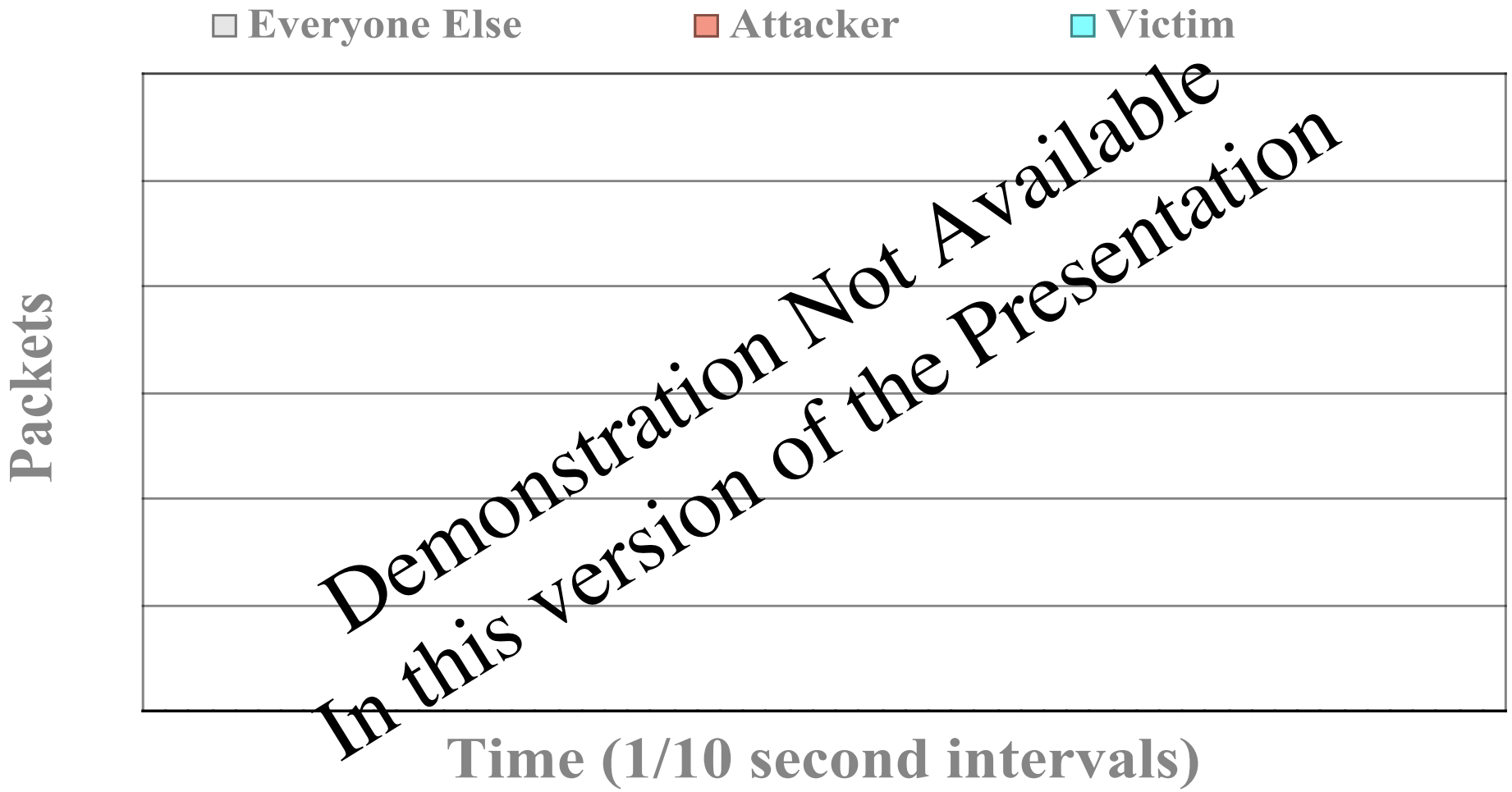
Department of Computer Science and Engineering
University of California, San Diego

Motivation

- 802.11-based networks have flourished
  - Home, business, health care, military, etc.

- Security is an obvious concern
  - Threats to confidentiality well understood and being addressed [WPA, 802.11i]
  - Threats to availability (denial-of-service) not widely appreciated & not being addressed

Live
802.11 DoS Demonstration



802.11 DoS Attacks

- RF Jamming
  - Real threat, 802.11 highly vulnerable; not our focus
- Bandwidth consumption (flooding)
  - 802.11 has same vulnerability as wired nets; not our focus

- **Attacks on 802.11 protocol itself**
  - Easy to mount, low overhead, selective, hard to debug
  - **Media access vulnerabilities**
  - **Management vulnerabilities**

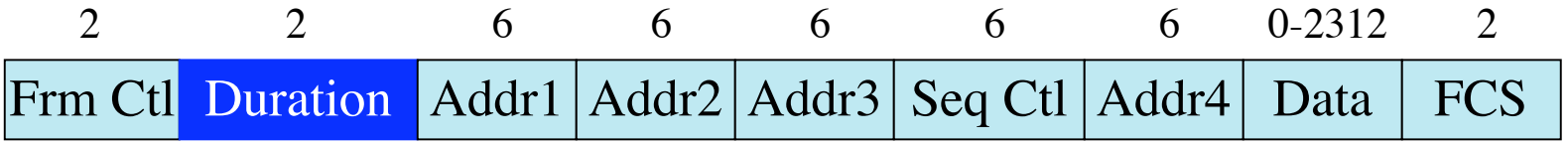
- **This talk focuses on these DoS attacks, their practicality, their effectiveness and how to defend against them**

Media Access Vulnerabilities

- 802.11 includes collision avoidance mechanisms
- Typically require **universal cooperation** between all nodes in the network
- Media access vulnerabilities arise from the assumption of universal cooperation

- Virtual carrier sense is an example of a media access mechanism that is vulnerable to DoS attacks

NAV Vulnerability

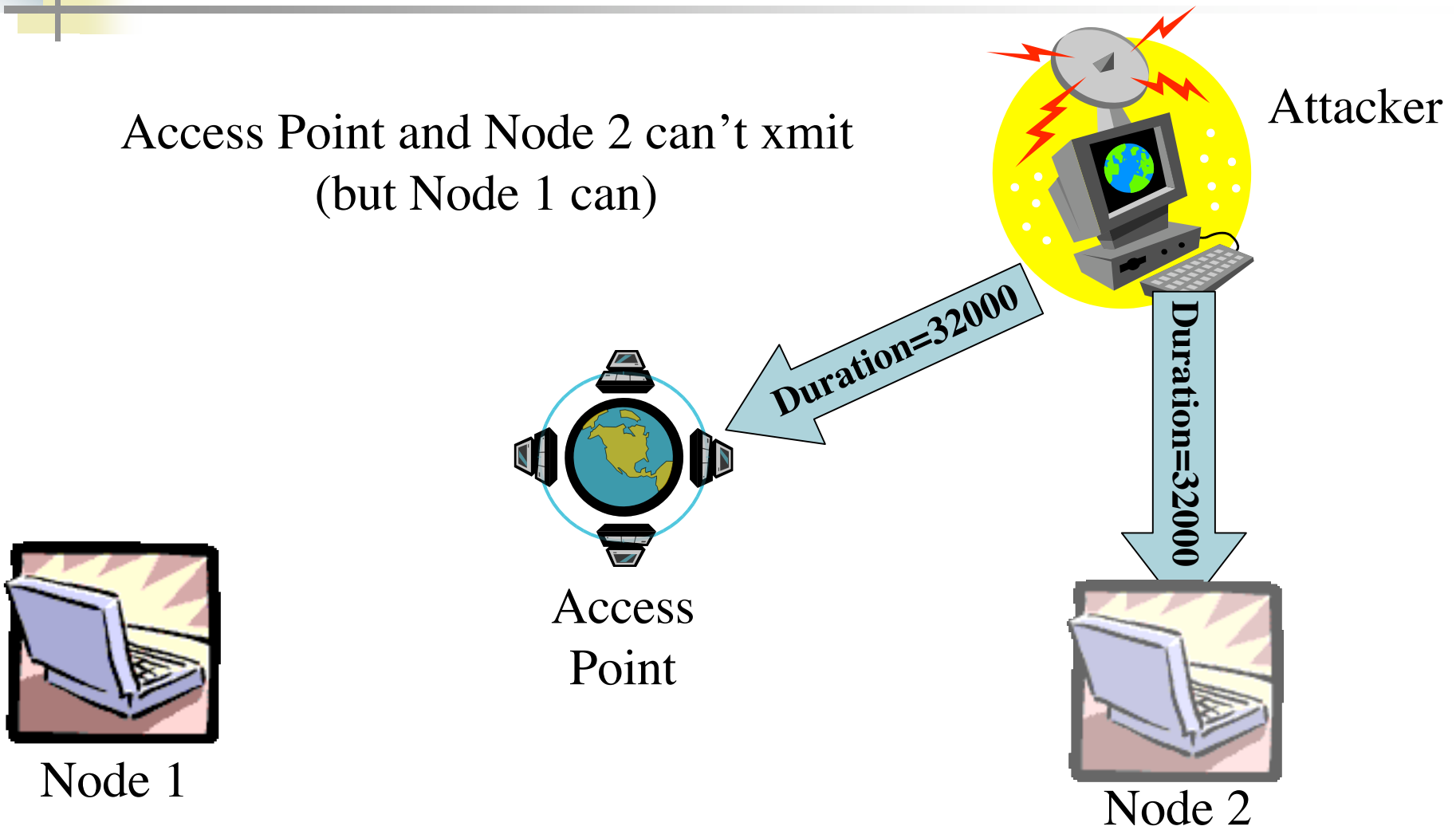


802.11 General Frame Format

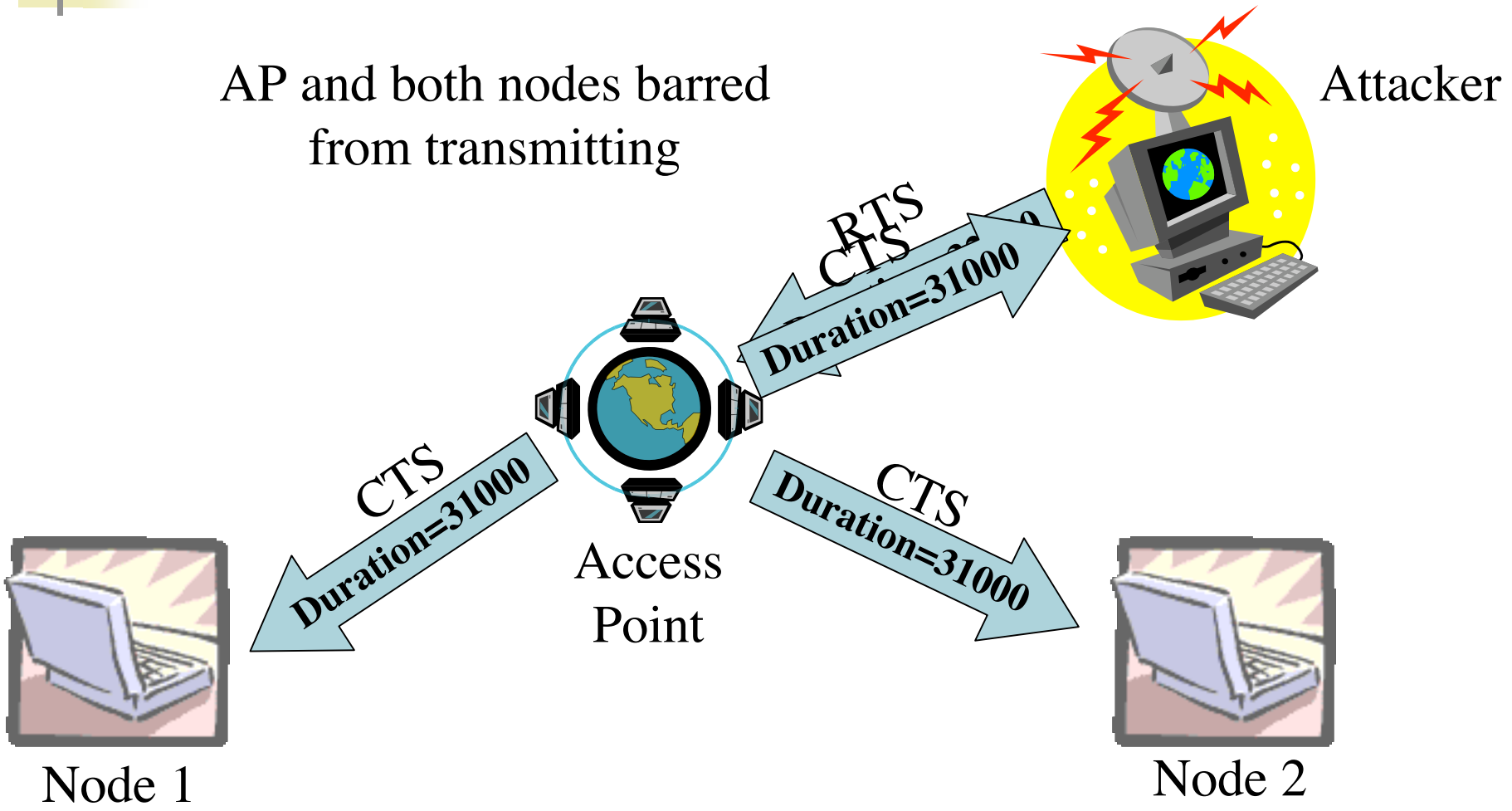
- Virtual carrier sense allows a node to reserve the radio channel
- Each frame contains a duration value
  - Indicates # of microseconds channel is reserved
  - Tracked per-node; Network Allocation Vector (NAV)
  - Used by RTS/CTS
- Nodes only allowed to xmit if NAV reaches 0

Simple NAV Attack: Forge packets with large Duration

Access Point and Node 2 can't xmit
(but Node 1 can)



Extending NAV Attack w/RTS



Conventional Wisdom

- NAV attack not a practical threat
  - Commodity hardware doesn't allow Duration field to be set
- But would be highly effective if implemented
  - Shut down all access to 802.11 network
- Both wrong…

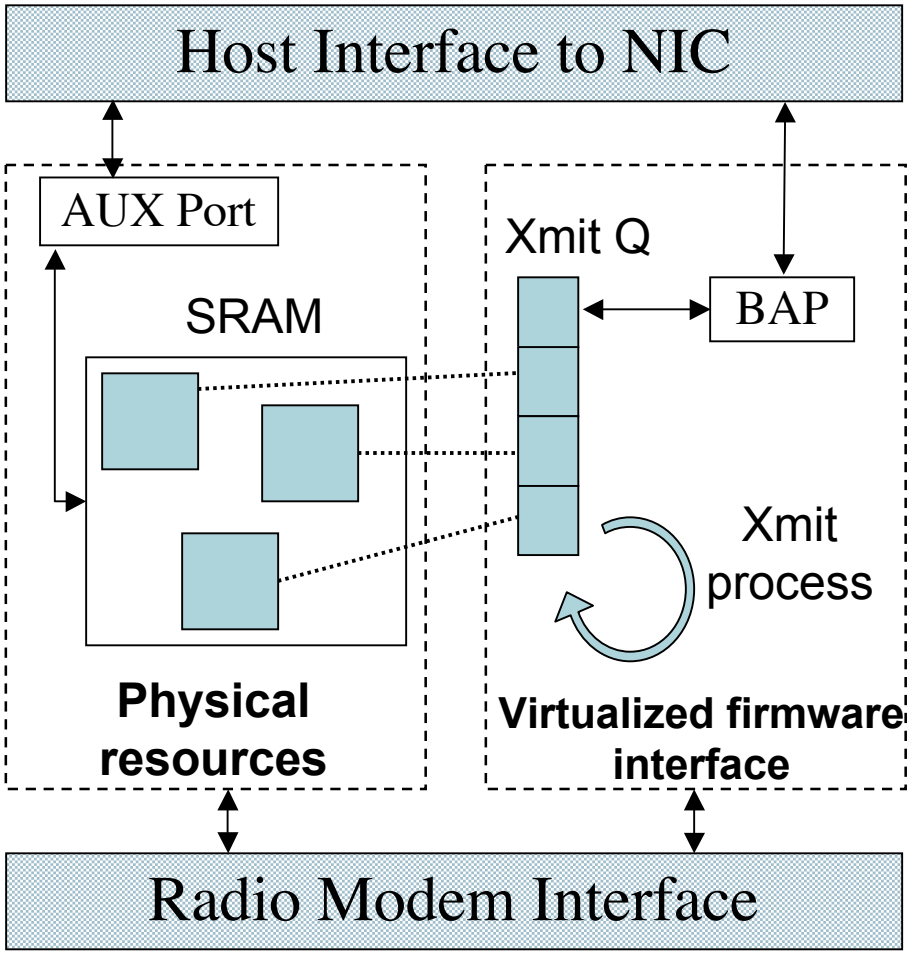
Commodity 802.11 hardware

- Firmware-driven microcontroller
  - Same code/architecture shared by most popular vendors (Choice Microsystems)
- Transmit path
  - Host provides frame to NIC and requests xmit
  - NIC firmware **validates** frame and **overwrites** key fields (e.g. duration) in real-time
  - Frame then sent to baseband radio interface
- **Not possible** to send arbitrary frames via firmware interface

How to Generate Arbitrary 802.11 Frames?

Key idea:
AUX/Debug Port allows Raw access to NIC SRAM

1. Download frame to NIC
2. Find frame in SRAM
3. Request transmission
4. Wait until firmware modifies frame
5. Rewrite frame via AUX port



Why the NAV attack doesn't work

- Surprise: many vendors do not implement the 802.11 spec correctly
- Duration field not respected by other nodes

Time (s)	Source	Destination	Duration (ms)	Type
1.294020		:e7:00:15:01	32.767	802.11 CTS
1.295192	:93:ea:e7:0f	:93:ea:ab:df	0.258	TCP Data
1.296540		:e7:0f	0	802.11 Ack
1.297869	:9	:e7:0f	0.258	TCP Data

1.2952 - 1.2940 = 1.2 ms

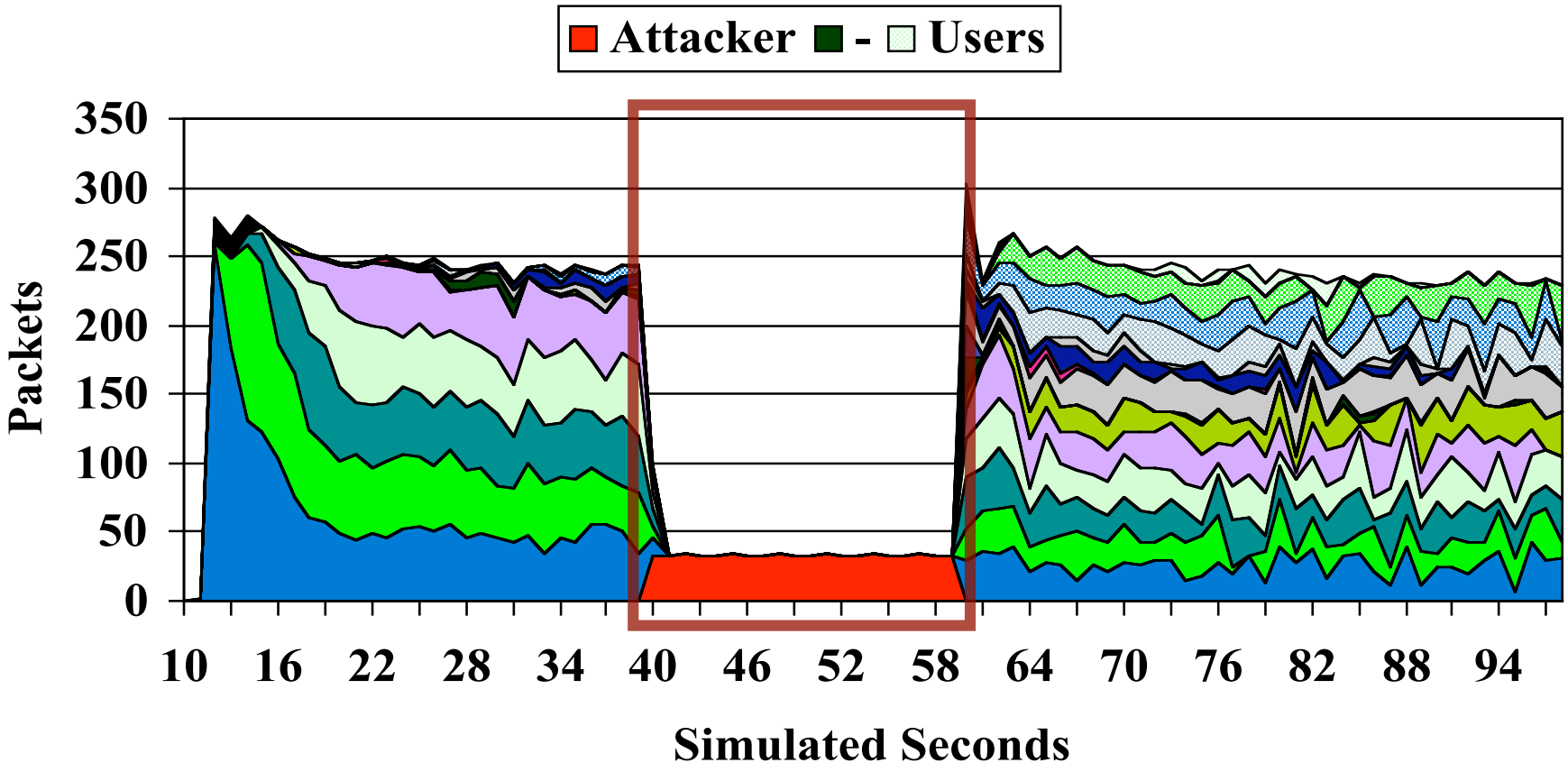
Excerpt from a NAV Attack Trace

Simulating the NAV attack

- This bug will likely get fixed
  - Valuable for 802.11-based telephony, video, etc.
- So how bad would the attack be?

- Simulated NAV attack using NS2
  - 18 Users
  - 1 Access Point
  - 1 Attacker
- 30 attack frames per second
- 32.767 ms duration per attack frame

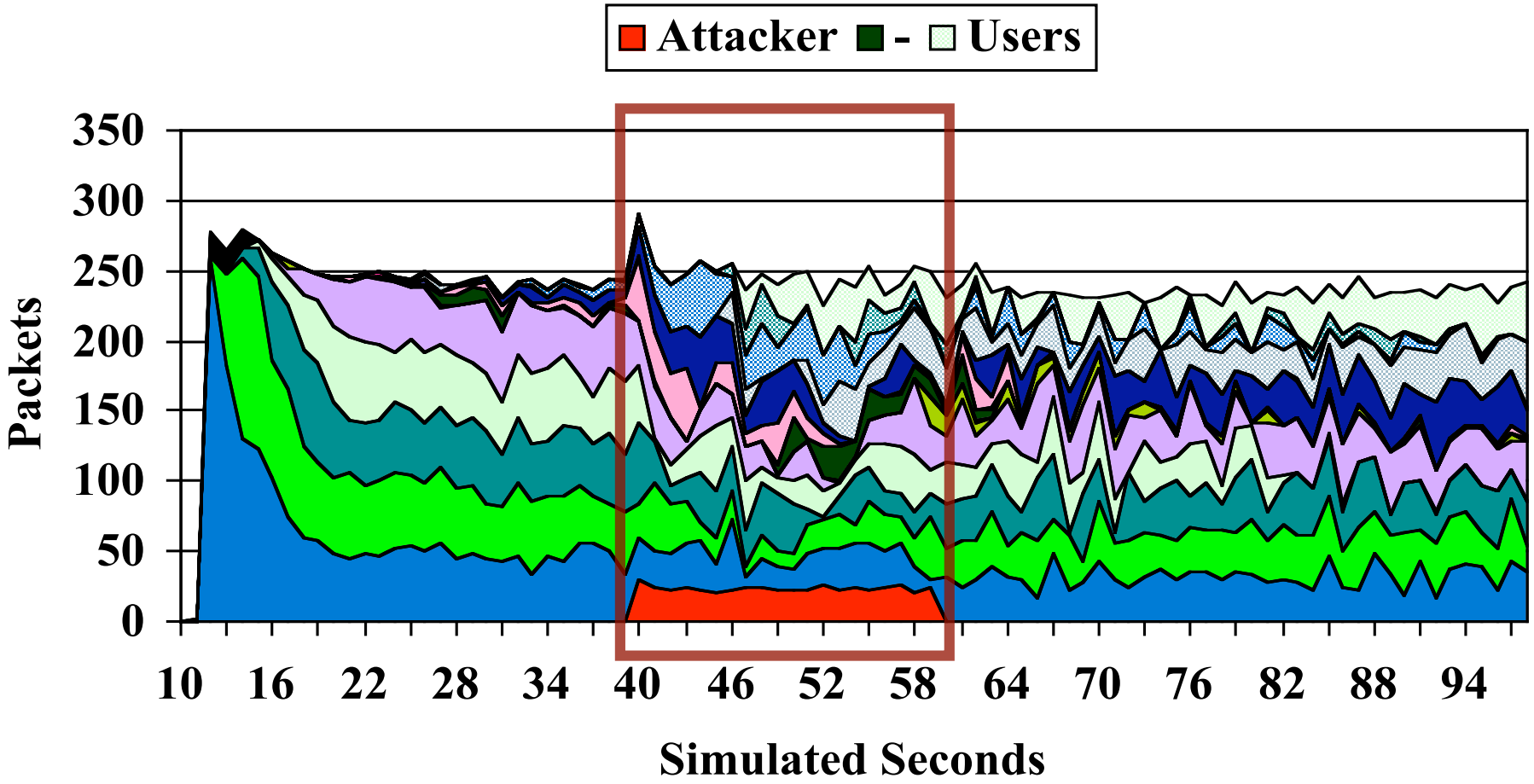
NAV Attack Simulation



Practical NAV Defense

- Legitimate duration values are relatively small
- Determine maximum *reasonable* NAV values for all frames
  - Each node enforces this limit
  - < .5 ms for all frames except ACK and CTS
  - ~3 ms for ACK and CTS
- Reran the simulation after adding defense to the simulator

Simulated NAV Defense

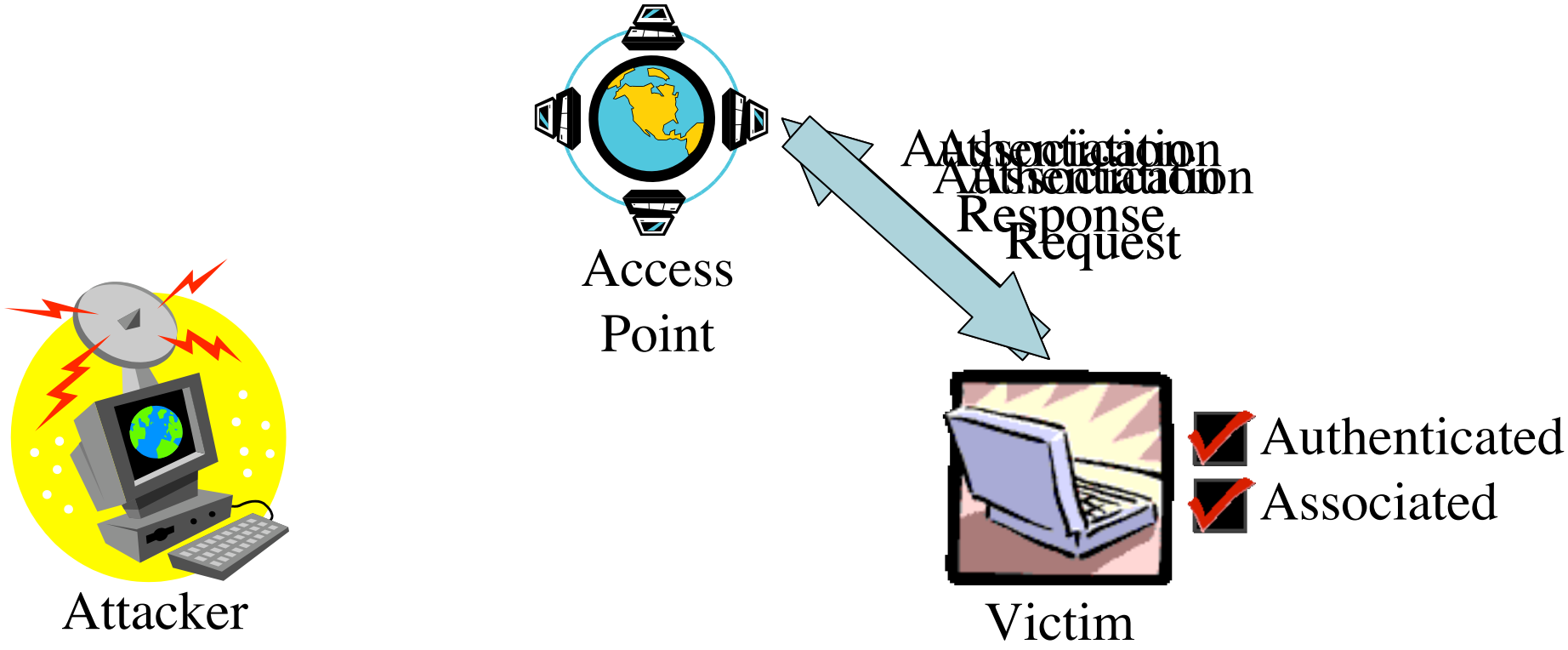


Management Vulnerabilities

- 802.11 Management functions
  - **Authentication** (validate identity)
  - **Association** (picking access point)
- Most management operations unprotected
  - **Easy** to spoof with false identity
  - Source of vulnerabilities
- This problem is not being fixed
  - Most management frames unencrypted
  - 802.1x ports allocated **after** management functions take place
  - 802.11i has deferred addressing this problem

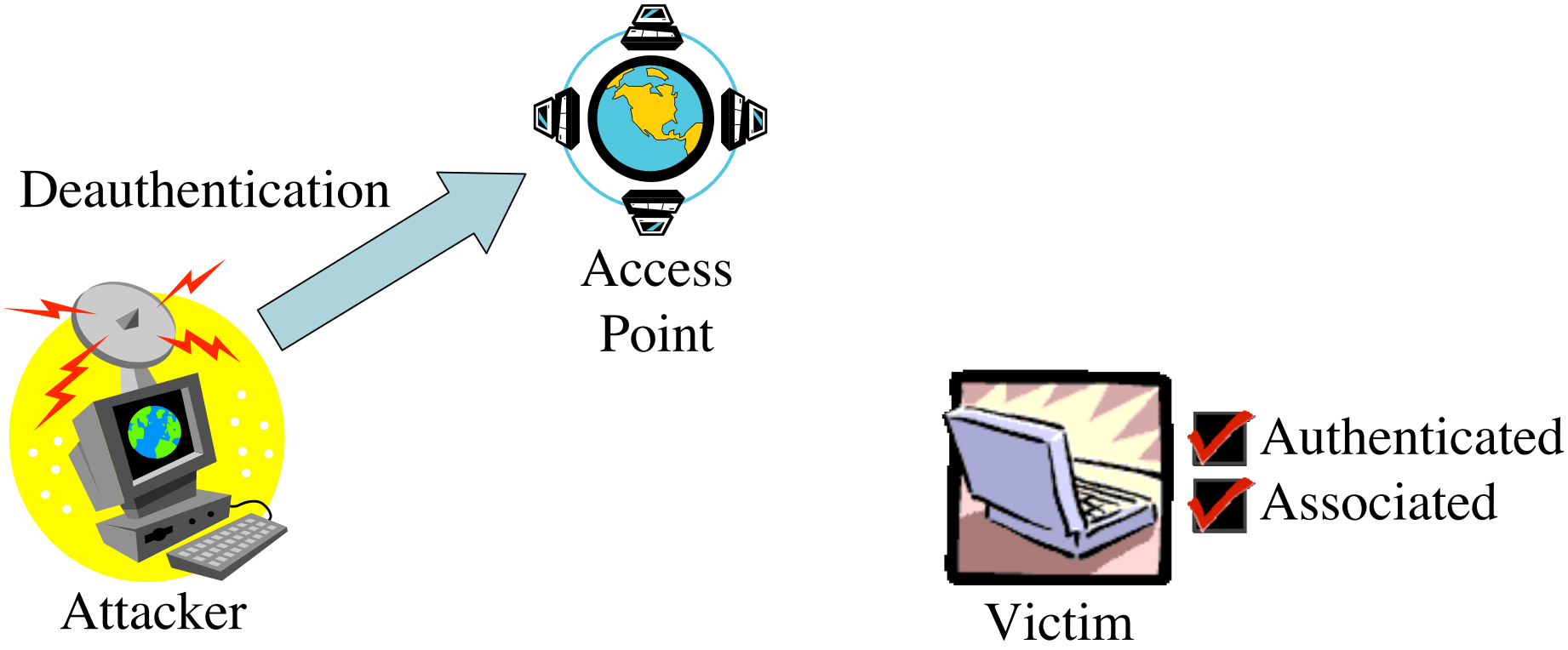
Deauth Attack

- 802.11 management requires nodes associate before sending data



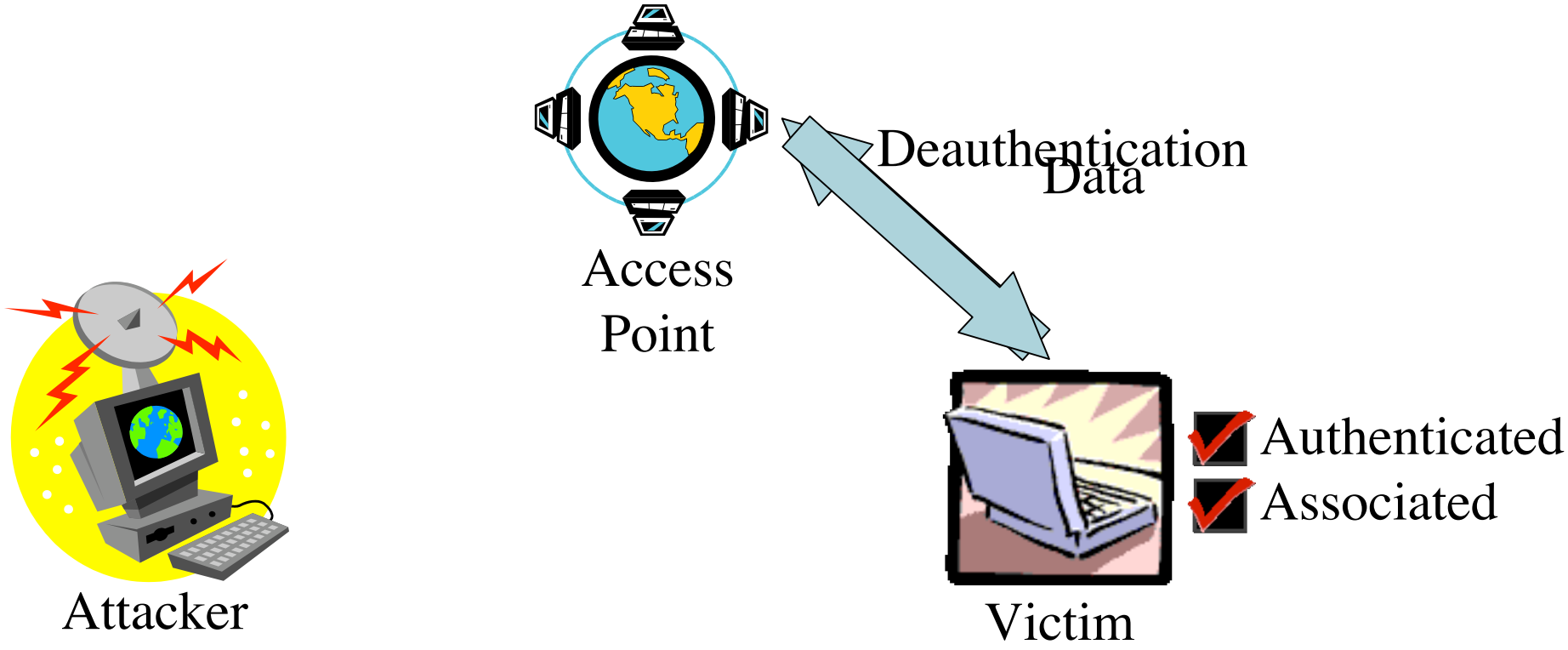
Deauth Attack

- Before node can transmit data, attacker send a spoofed deauthentication frame

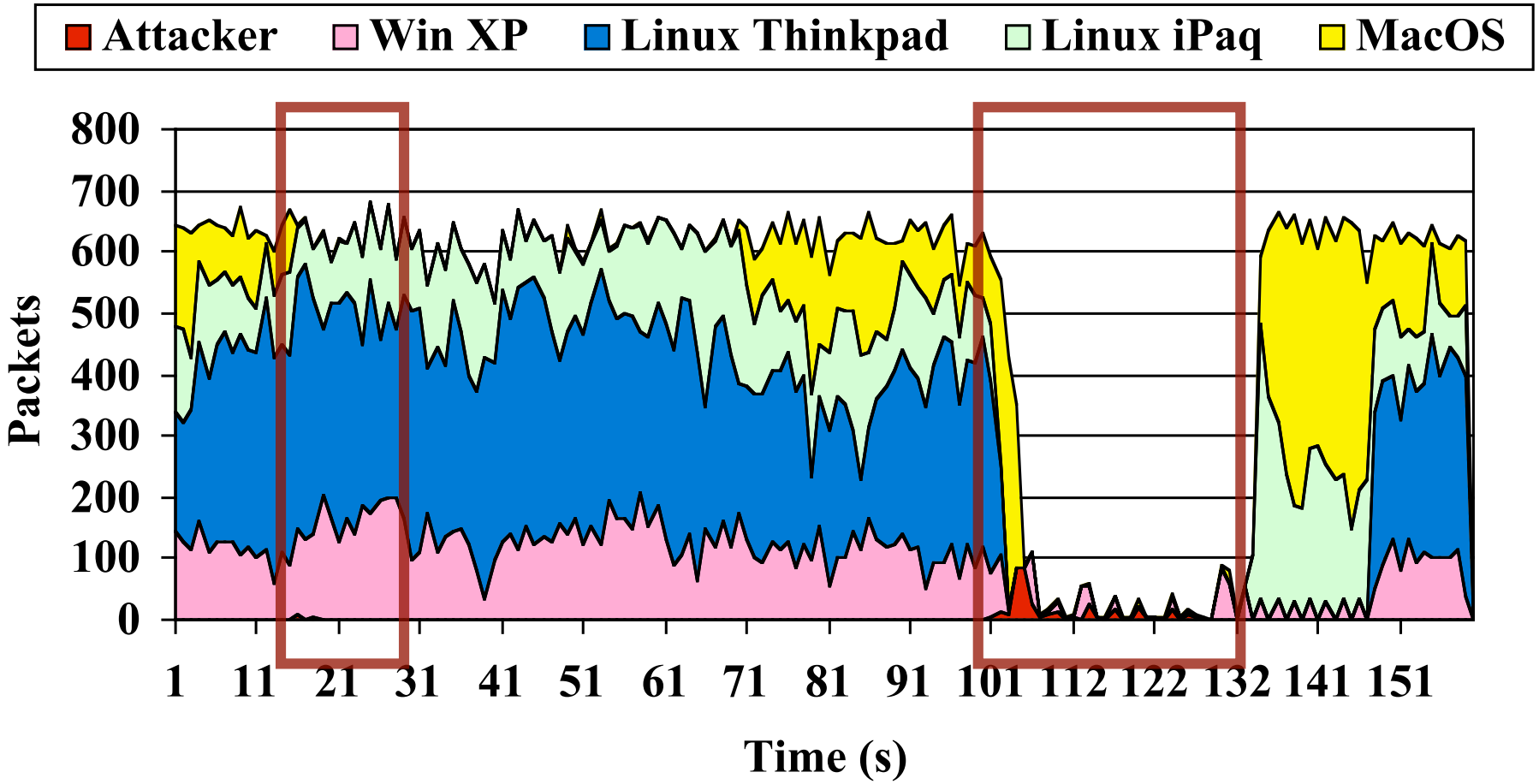


Deauth Attack

- Node attempts to transmit data, but it can not



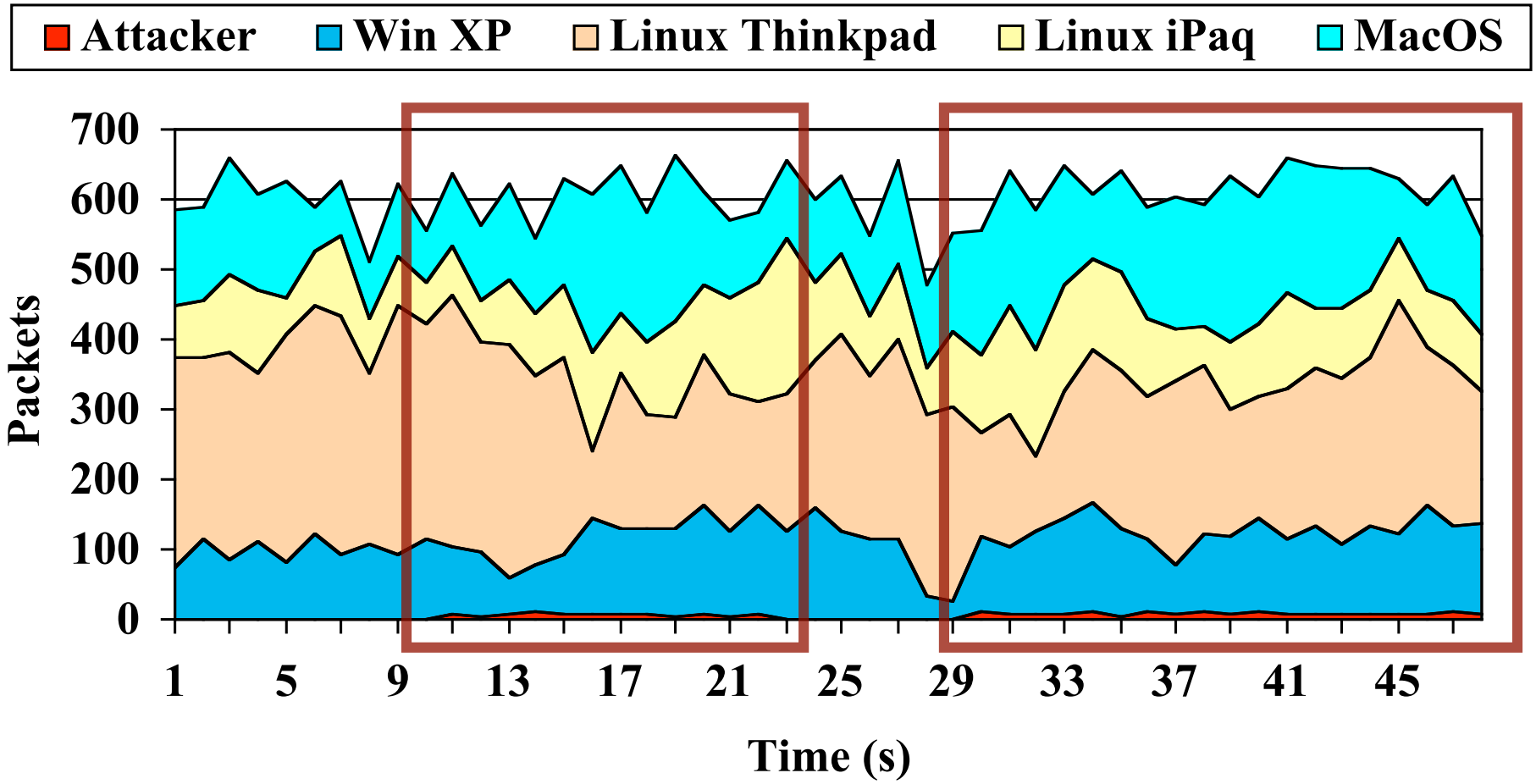
Deauth Attack Results



Practical Deauth Defense

- Based on the observed behavior that legitimate nodes do not deauthenticate themselves and then send data
- Delay honoring deauthentication request
  - Small interval (5-10 seconds)
  - If no other frames received from source then honor request
  - If source sends other frames then discard request
- Requires no protocol changes and is backwards compatible with existing hardware

Deauth Defense Results



Conclusion

- 802.11 DoS attacks require more attention
  - Easy to mount and not addressed by existing standards
- Should not depend on restricted firmware interfaces (can send arbitrary 802.11 pkts)
- Deauthentication attack is most immediate concern
  - Simple, practical defense shown to be effective

Hands-on Demonstration

- Attack implemented on an iPaq
- See me for a hands-on demonstration during the break