Framing Frames: Bypassing Wi-Fi Encryption by Manipulating Transmit Queues

Domien Schepers, ***Aanjhan Ranganathan***, Mathy Vanhoef

Northeastern University Khoury College of Computer and Information Sciences

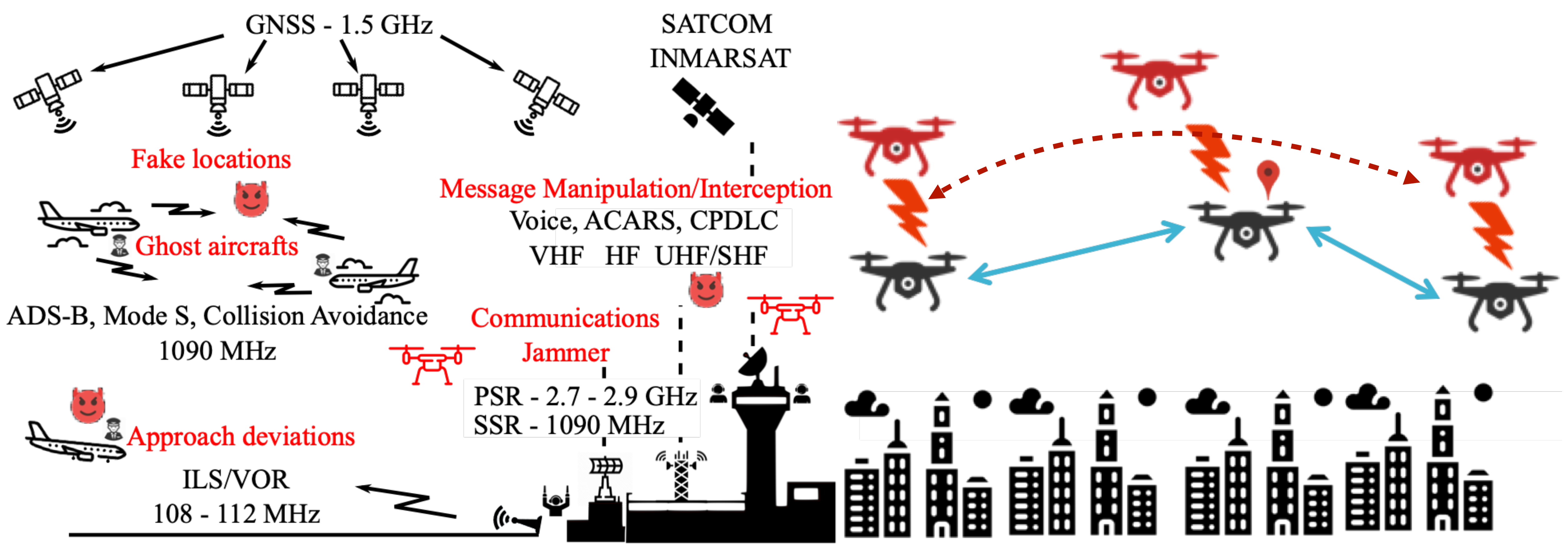
KU LEUVEN

WAC6 (colocated with CRYPTO 2023)

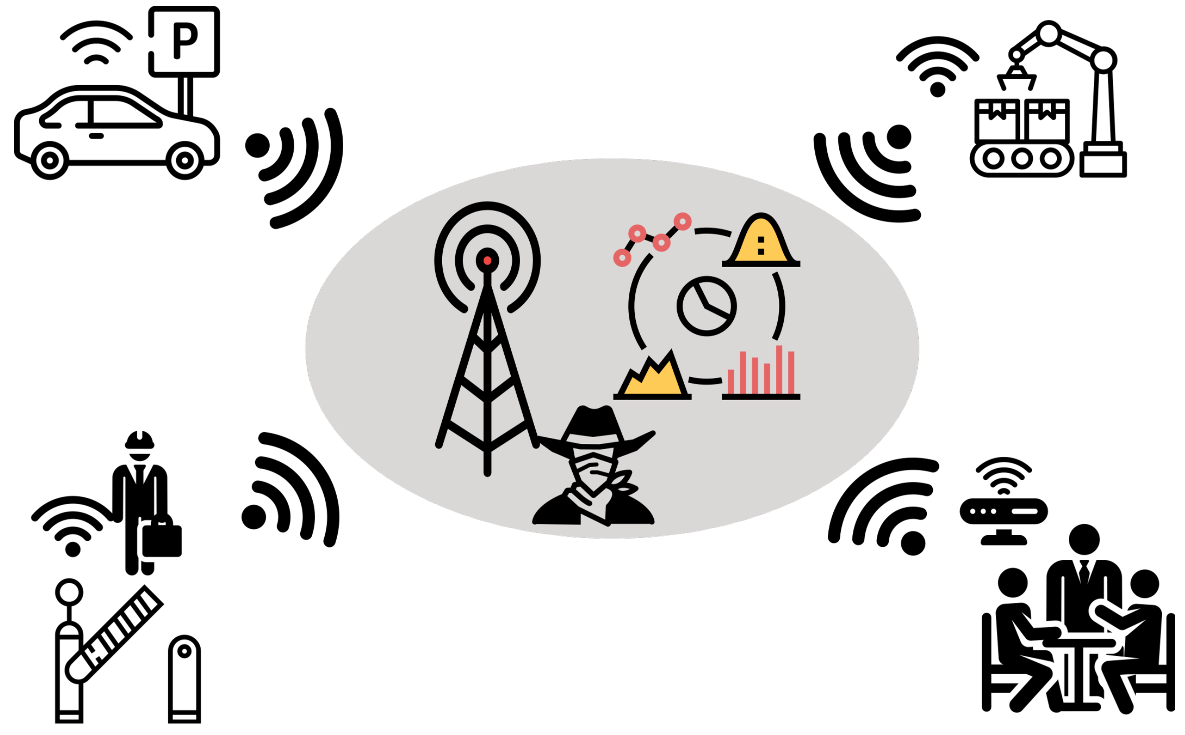
Signal Intelligence Lab @ Northeastern

Security and privacy of wireless networks with a focus on *autonomous cyber-physical systems and smart ecosystems*.

Secure and Private Wide-area Positioning



Aviation and Aerospace Security



Security and Privacy of xIoT

Faculty

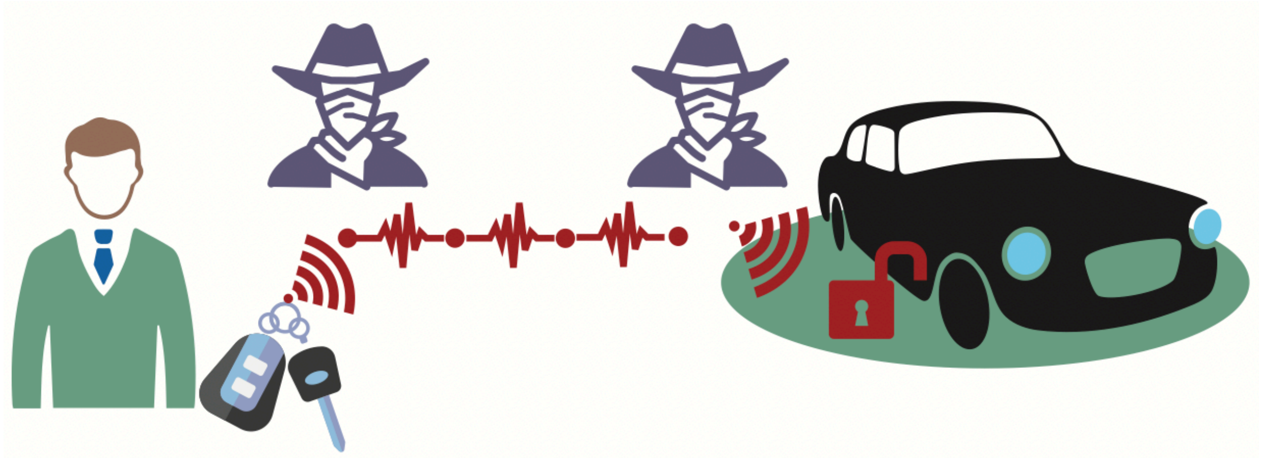
Aanjhan Ranganathan
Assistant Professor
www.aanjhan.com

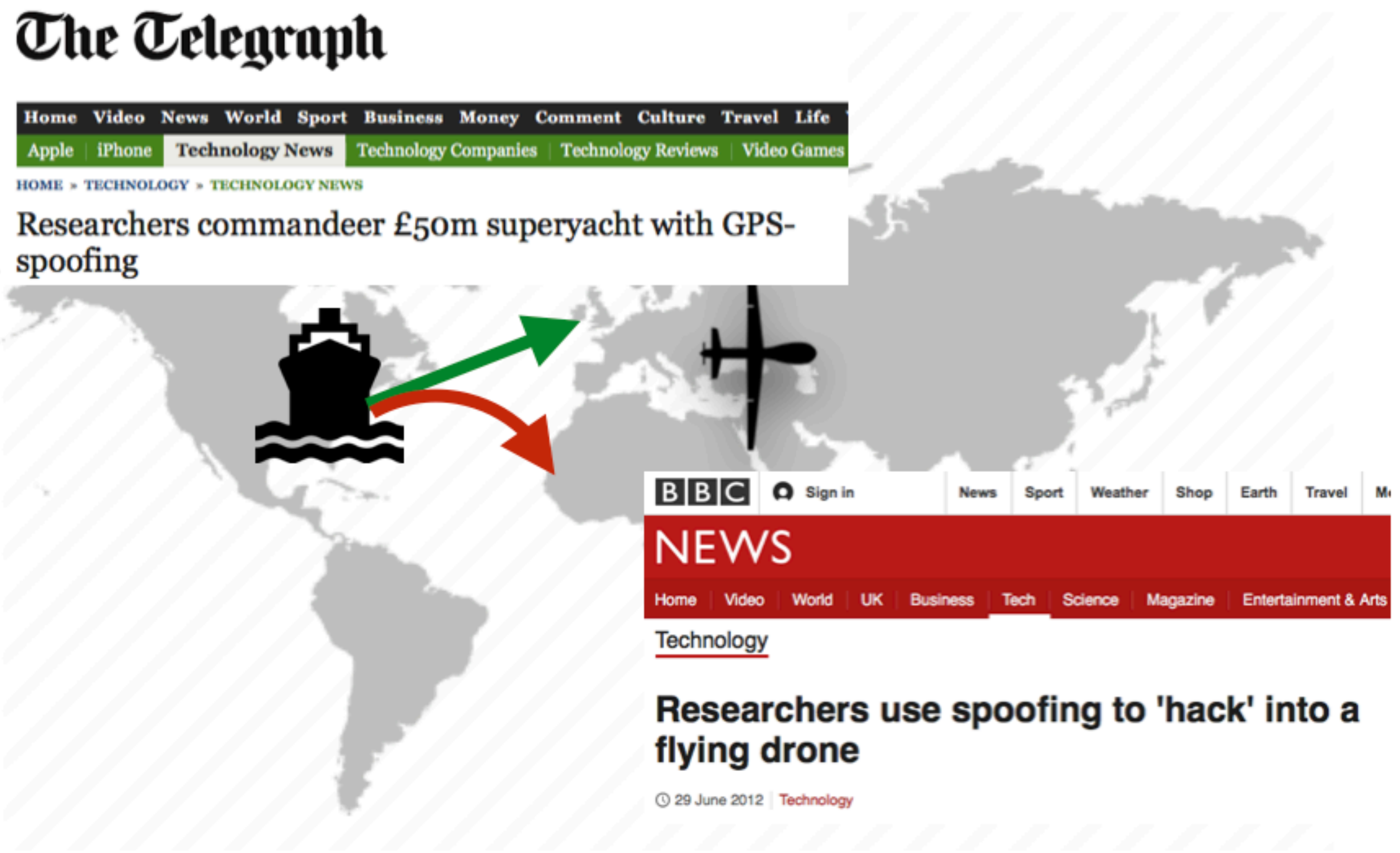
PhD Students

Secure Proximity and Location Verification

Towards Secure and Private Wide-area Positioning

Attacks on Location




The Telegraph

Home Video News World Sport Business Money Comment Culture Travel Life
Apple iPhone Technology News Technology Companies Technology Reviews Video Games
HOME » TECHNOLOGY » TECHNOLOGY NEWS

Researchers commandeer £50m superyacht with GPS-spoofing

BBC NEWS

Sign in News Sport Weather Shop Earth Travel M
Home Video World UK Business Tech Science Magazine Entertainment & Arts

Technology

Researchers use spoofing to 'hack' into a flying drone

29 June 2012 | Technology

Selected Research

An Experimental Study of GPS Spoofing and Takeover Attacks on UAVs, Harshad Sathaye, Martin Strohmeier, Vincent Lenders, Aanjhan Ranganathan (USENIX Security 2022)

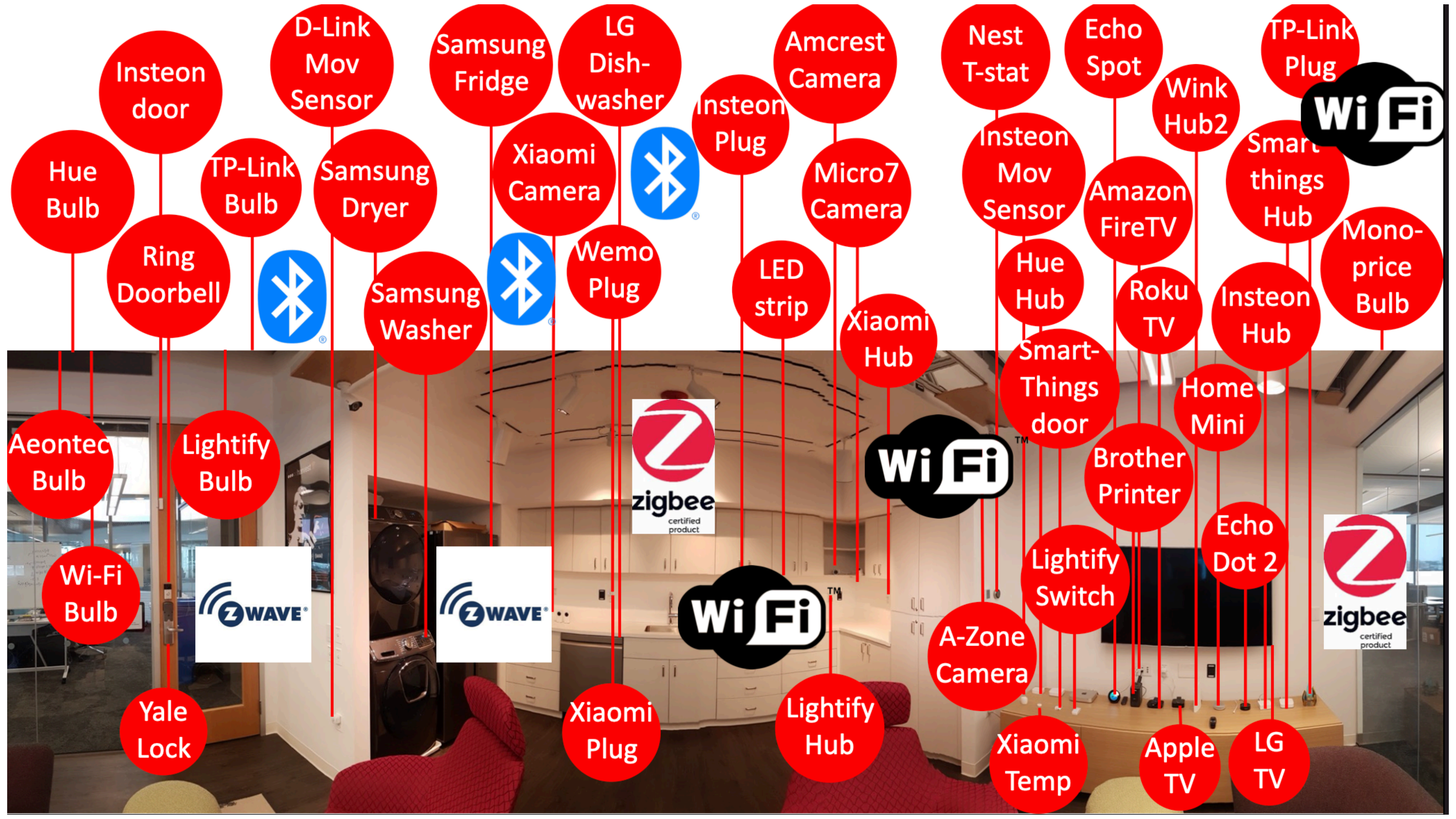
VRange: Enabling Secure Ranging in 5G-NR Wireless Networks, Mridula Singh, Marc Roeschlin, Aanjhan Ranganathan, Srdjan Capkun (NDSS 2022)

SemperFi: Anti-spoofing GPS receiver for UAVs, Harshad Sathaye, Gerald LaMountain, Pau Closas, Aanjhan Ranganathan (NDSS 2022)

Wireless Attacks on Aircraft Instrument Landing Systems, Harshad Sathaye, Domien Schepers, Aanjhan Ranganathan, Guevara Noubir (USENIX Security 2019)

Security and Privacy in xIoT

Validating and Building Trustworthy Smart Ecosystems



Mon(Iot)Or Lab at Northeastern University

Selected Research

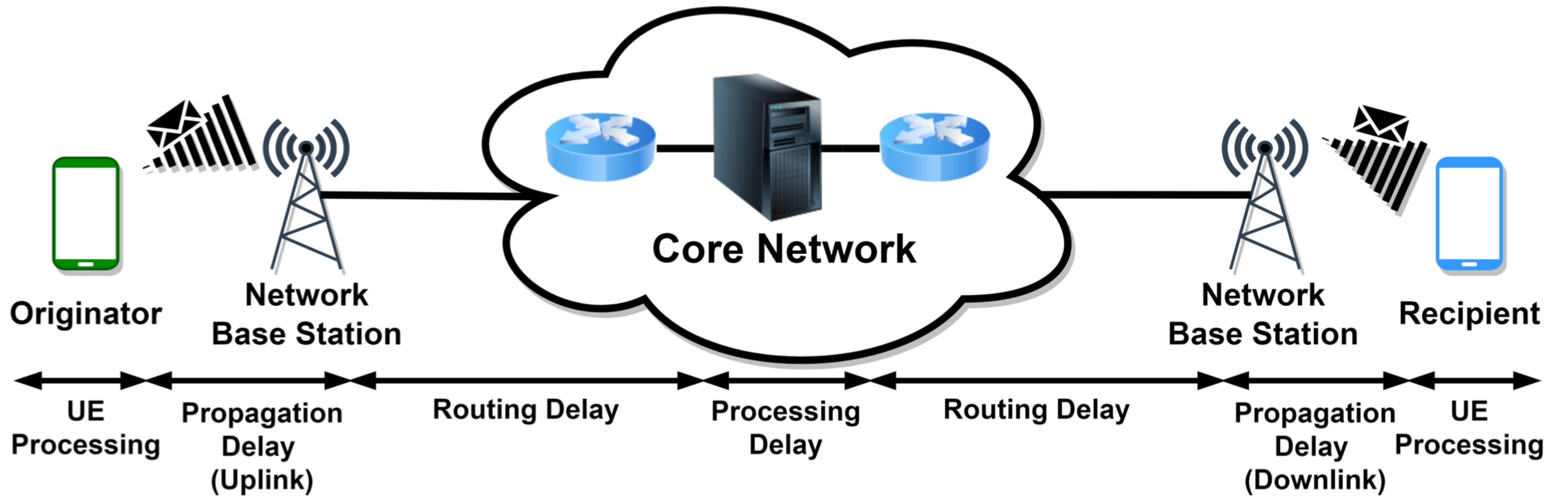
Track You: A Deep Dive into Safety Alerts for Apple AirTags, Narmeen Shafqat, Nicole Gerzon, Maggie Von Nortwick, Victor Sun, Alan Mislove, Aanjhan Ranganathan (PETS 2023)

ZLeaks: Passive Inference Attacks on Zigbee based Smart Homes, Narmeen Shafqat, Daniel Dubois, Dave Choffnes, Aaron Schulman, Dinesh Bharadia, Aanjhan Ranganathan (ACNS 2022, *Best Student Paper Award*)

Privacy-Preserving Positioning in Wi-Fi Fine Timing Measurements, Domien Schepers, Aanjhan Ranganathan (PETS 2022)

I Send, Therefore I Leak: Information Leakage in Low-Power Wide Area Networks, Patrick Leu, Ivan Puddu, Aanjhan Ranganathan, Srdjan Capkun (WiSec 2018)

Wi-Fi and Cellular Security



Selected Research

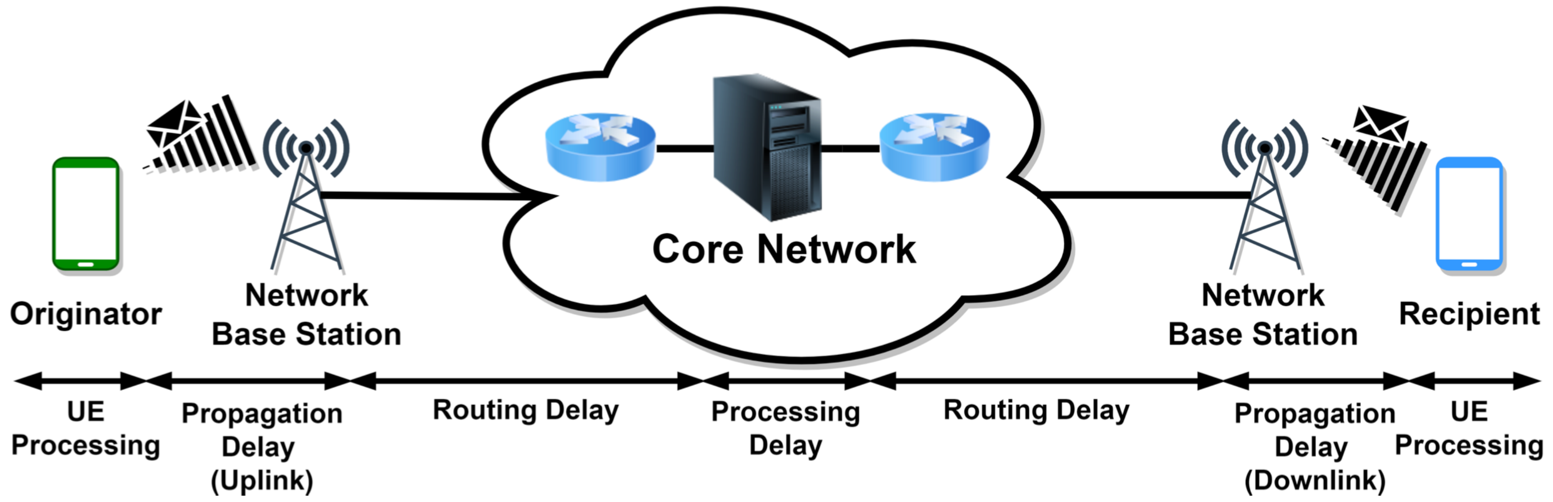
Framing Frames: Bypassing Wi-Fi Encryption by Manipulating Transmit Queues
Domien Schepers, Aanjhan Ranganathan, Mathy Vanhoef (USENIX Security 2023)

Freaky Leaky SMS: Extracting User Locations by Analyzing SMS Timings

Evangelos Bitsikas, Theo Schnitzler, Christina Poepper, Aanjhan Ranganathan (USENIX Security 2023)

On the Robustness of Wi-Fi Deauthentication Countermeasures, Domien Schepers, *Aanjhan Ranganathan*, Mathy Vanhoef (Wisec 2022)

Wi-Fi and Cellular Security



Selected Research

Framing Frames: Bypassing Wi-Fi Encryption by Manipulating Transmit Queues
Domien Schepers, Aanjhan Ranganathan, Mathy Vanhoef (USENIX Security 2023)

Freaky Leaky SMS: Extracting User Locations by Analyzing SMS Timings

Evangelos Bitsikas, Theo Schnitzler, Christina Poepper, Aanjhan Ranganathan (USENIX Security 2023)

On the Robustness of Wi-Fi Deauthentication Countermeasures, Domien Schepers, *Aanjhan Ranganathan*, Mathy Vanhoef (Wisec 2022)

History of Wi-Fi

- WEP (1999): quickly broken [FMS01]
- WPA1/2 (~2003)
  - Offline password brute-force
  - **KRACK** & **Kraken** [VP17,VP18]
- WPA3 (2018):
  - **Dragonblood** side-channels [VR20]

https://www.eset.com/int/kr00k

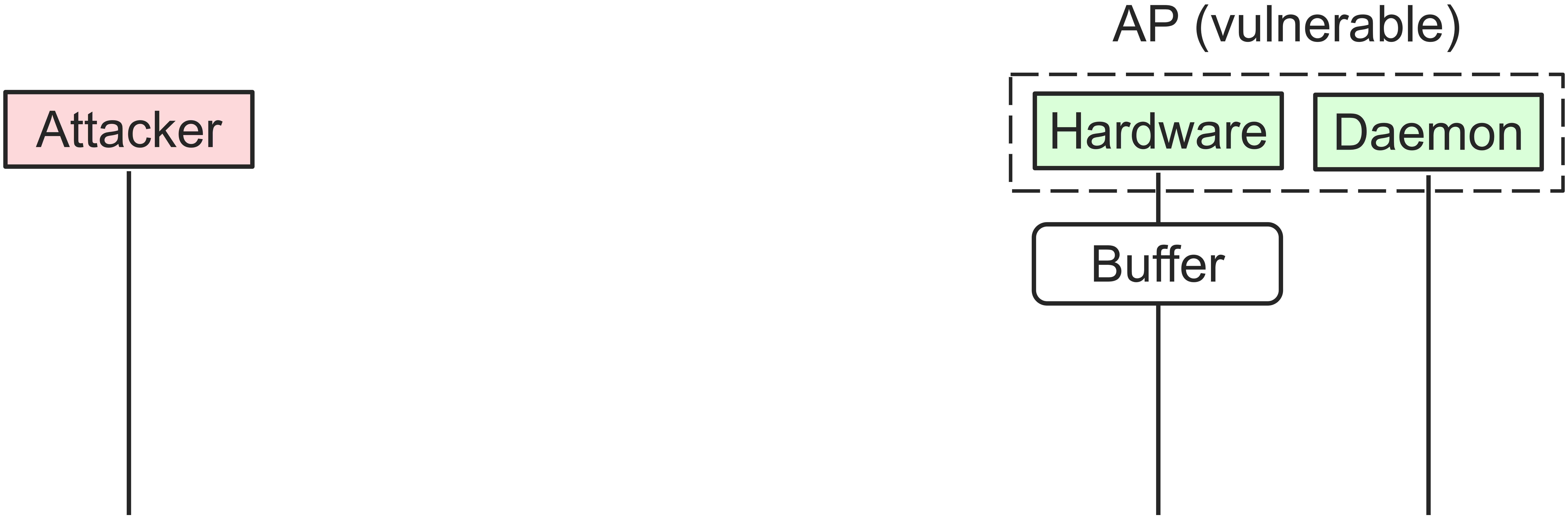
https://www.krackattacks.com

https://wpa3.mathyvanhoef.com

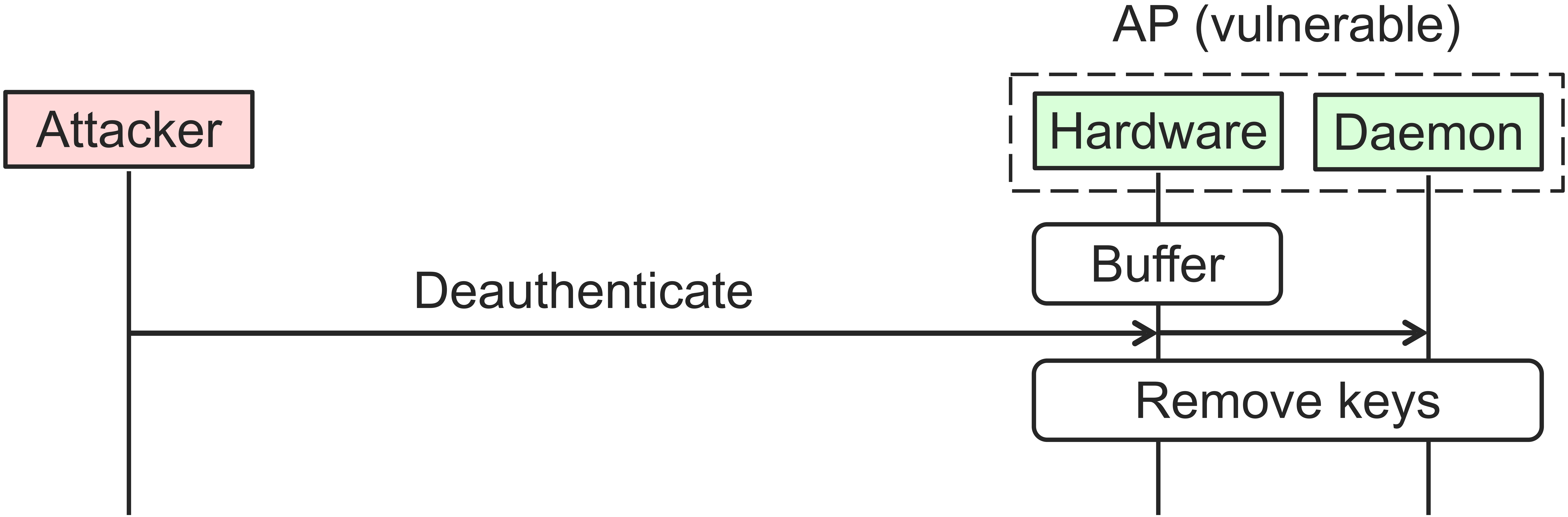
https://www.fragattacks.com

Background: Kr00k implementation flaw

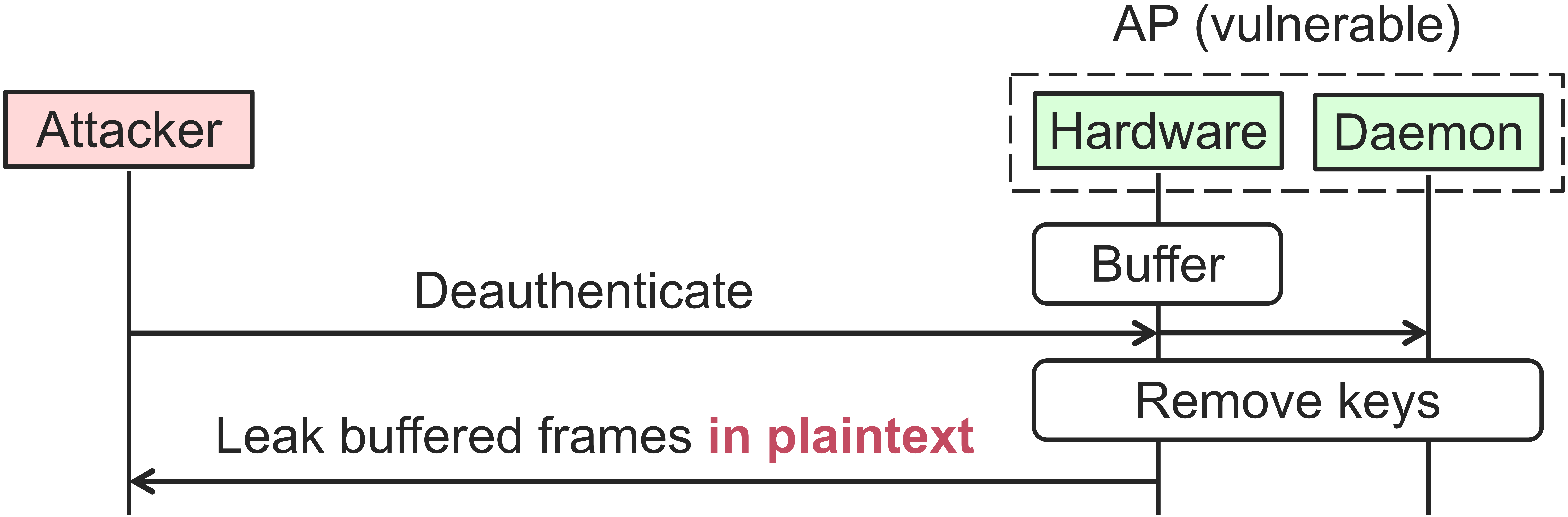
Background: Kr00k implementation flaw



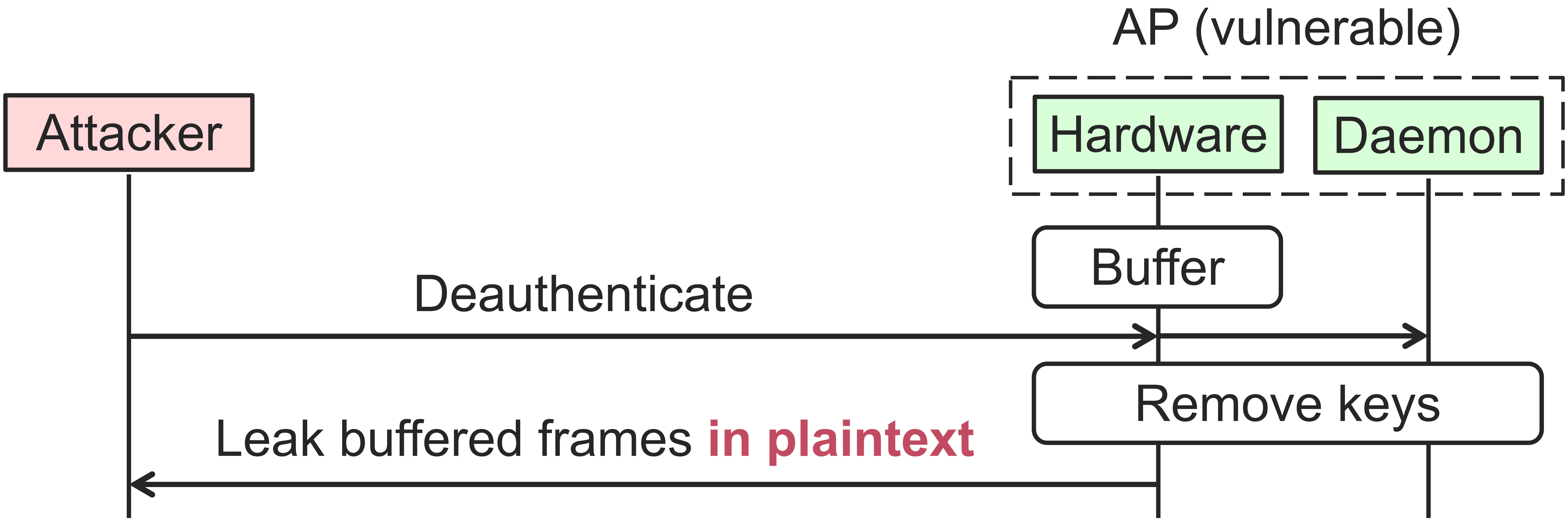
Background: Kr00k implementation flaw



Background: Kr00k implementation flaw



Background: Kr00k implementation flaw



Research question: how are security contexts managed?

The Security Context

Formally known as the *‘security association’* in the IEEE 802.11 standard:

- Protocol suites, negotiated encryption keys, packet counters, …
- All information needed to securely communicate.

The Security Context

Formally known as the *‘security association’* in the IEEE 802.11 standard:

- Protocol suites, negotiated encryption keys, packet counters, …
- All information needed to securely communicate.

What is the relation between security context and frames in the transmit queues?

- What happens to a queue if the security context changes?
  E.g., reconnection.

1. Can an Adversary Manipulate the Queue and Security Context?

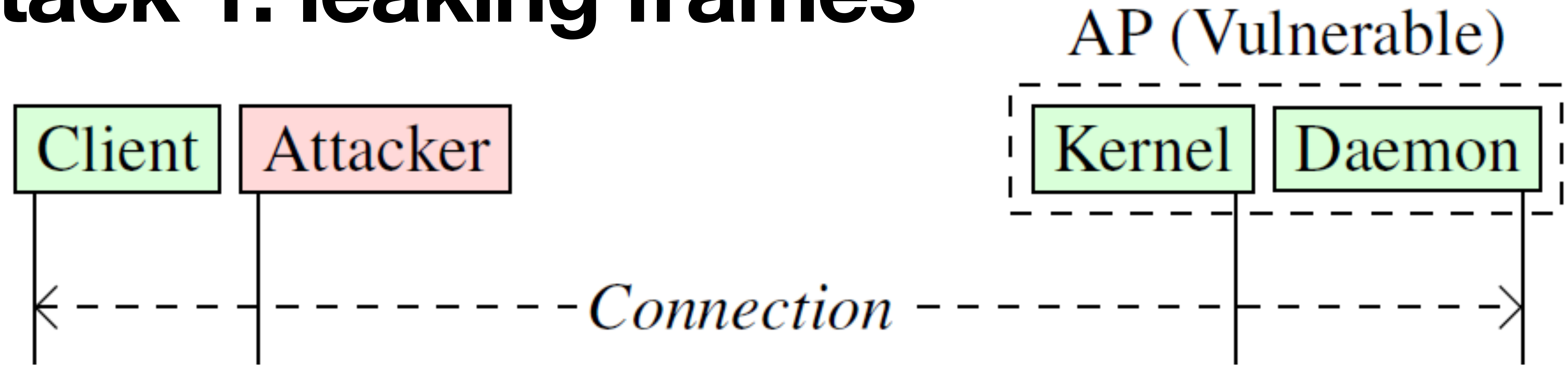
1. Can an Adversary Manipulate the Queue and Security Context?
2. What are the implications?

Finding 1: Leaking Frames

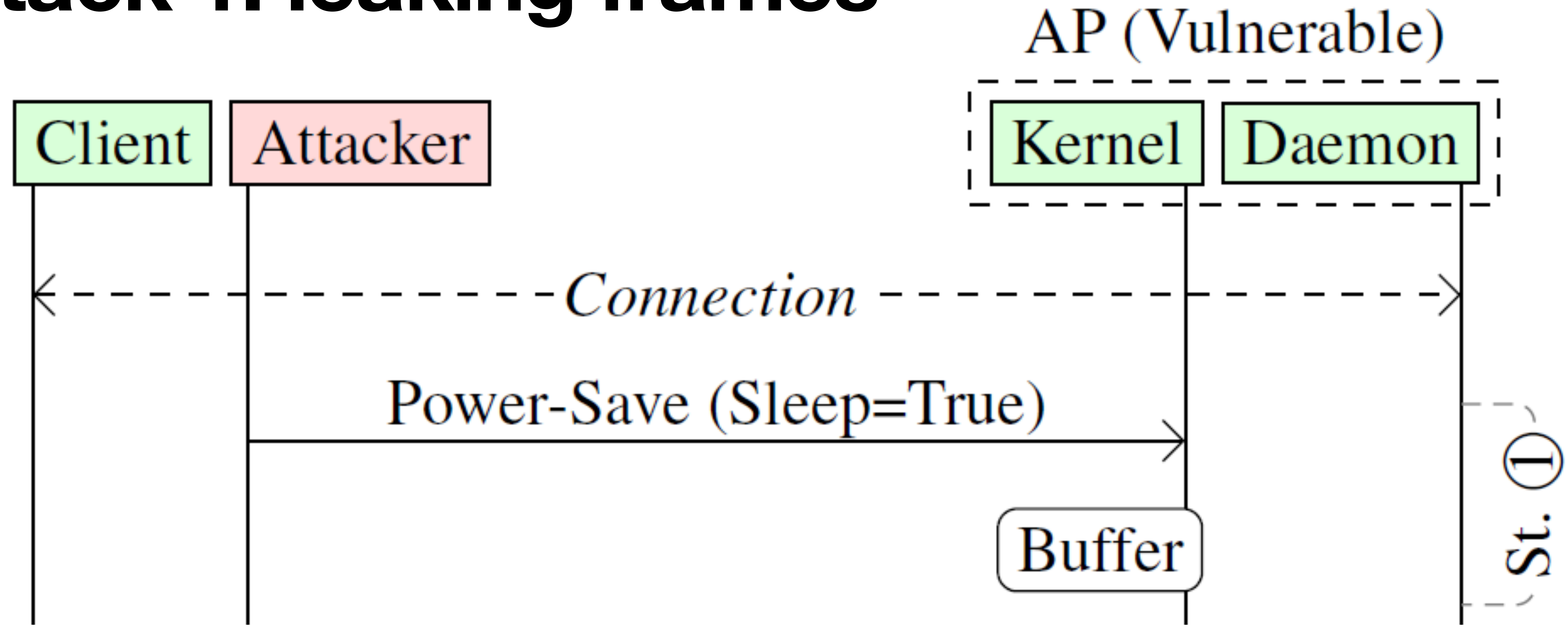
Attack 1: leaking frames

AP (Vulnerable)
Client
Attacker
Kernel
Daemon

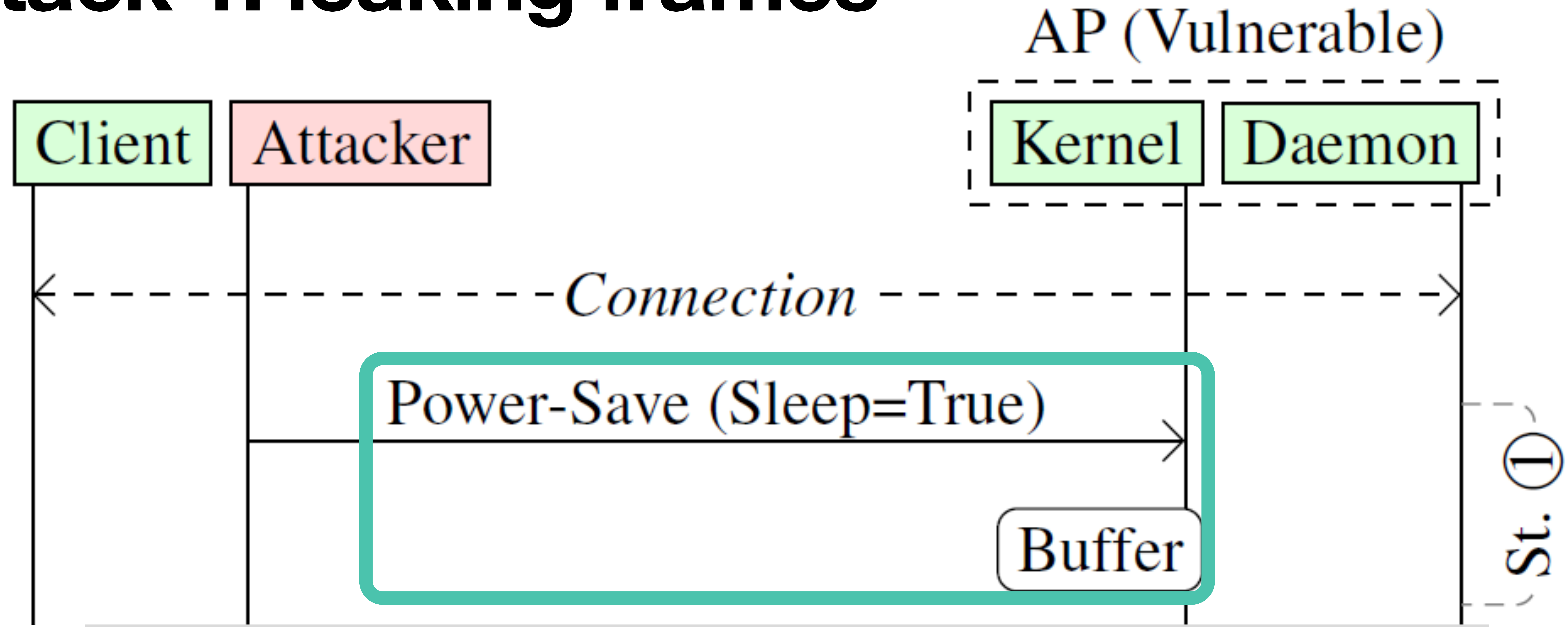
Attack 1: leaking frames



Attack 1: leaking frames

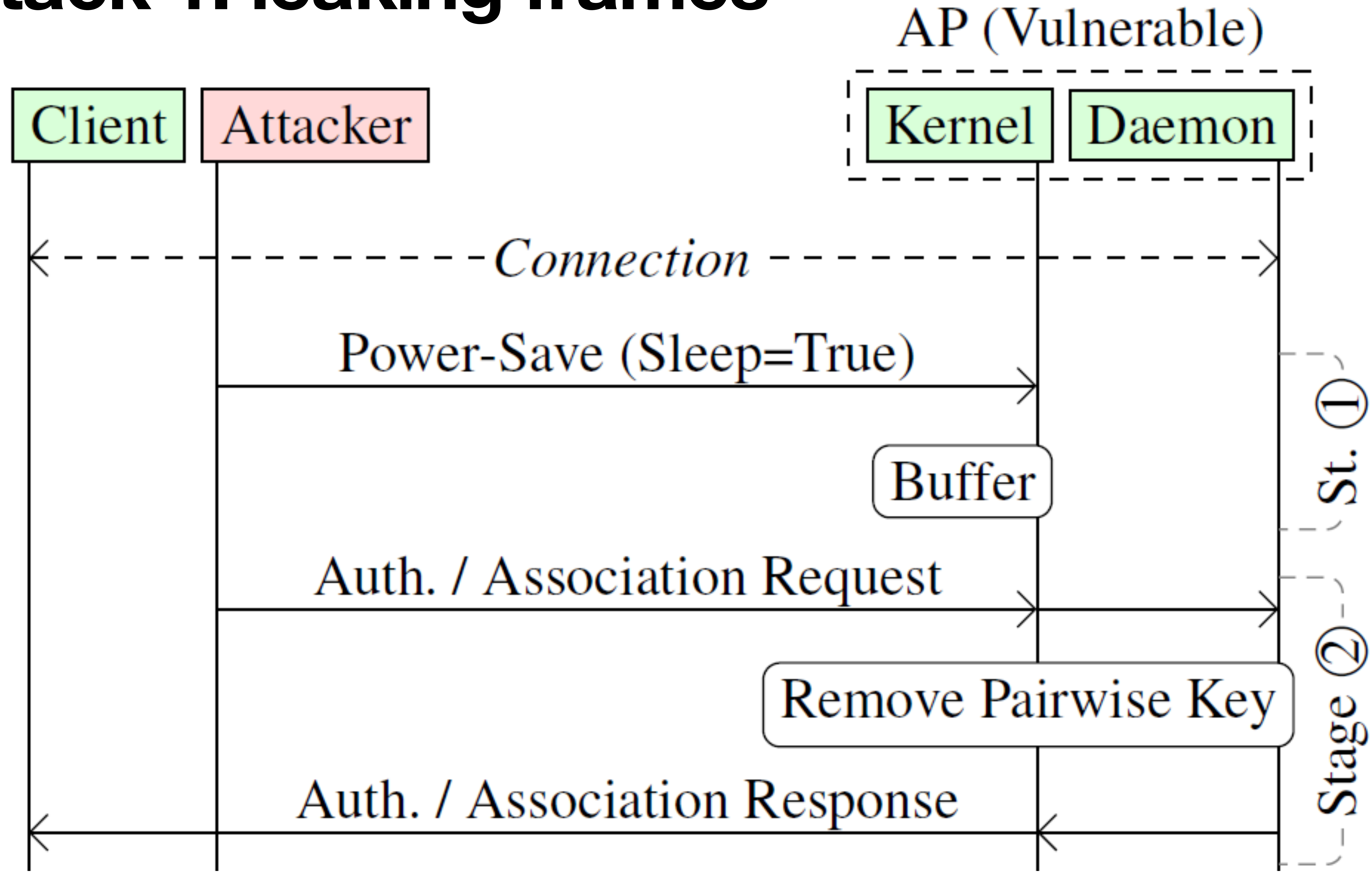


Attack 1: leaking frames

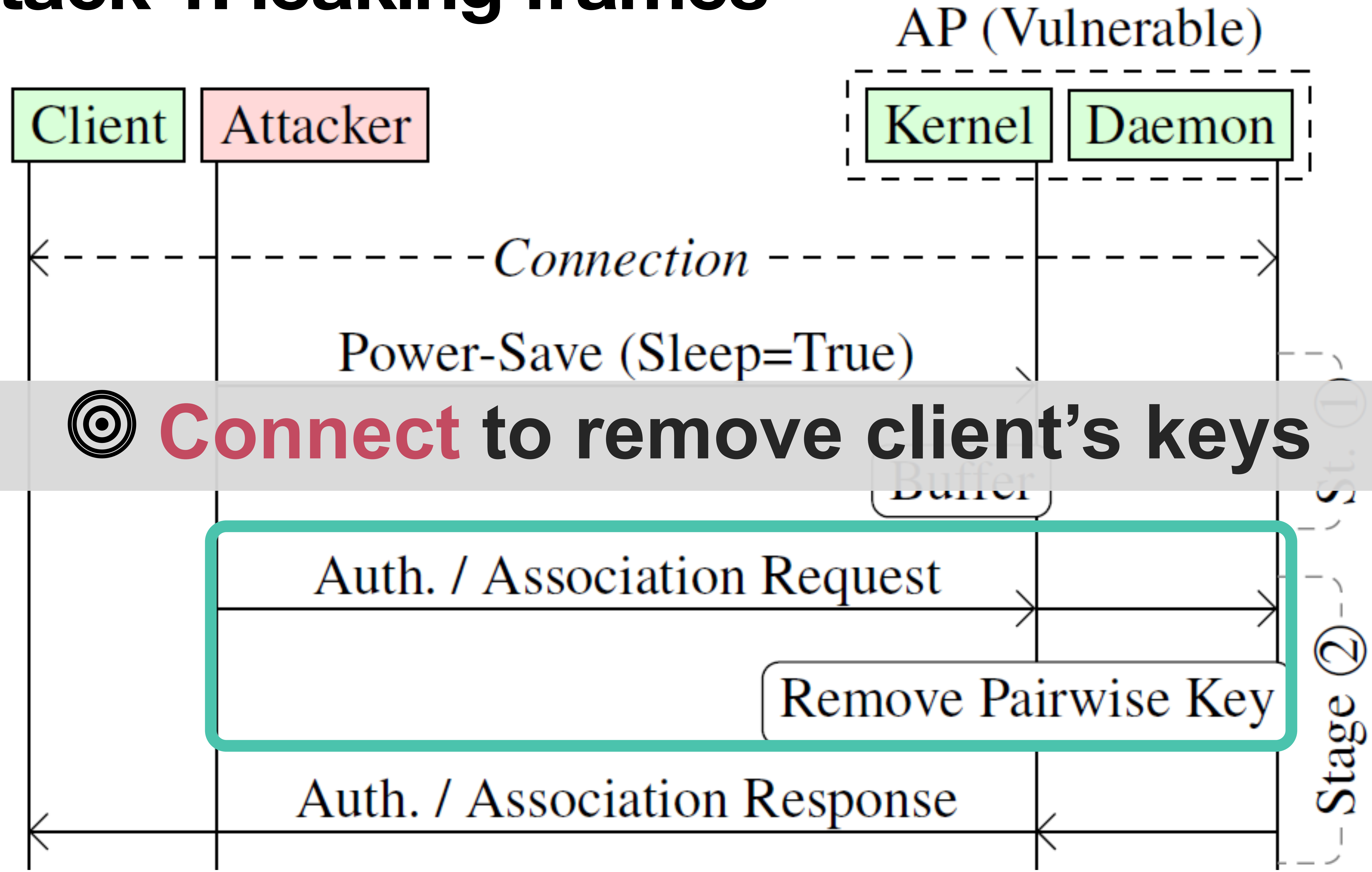


◎ **Controlled buffering**

Attack 1: leaking frames

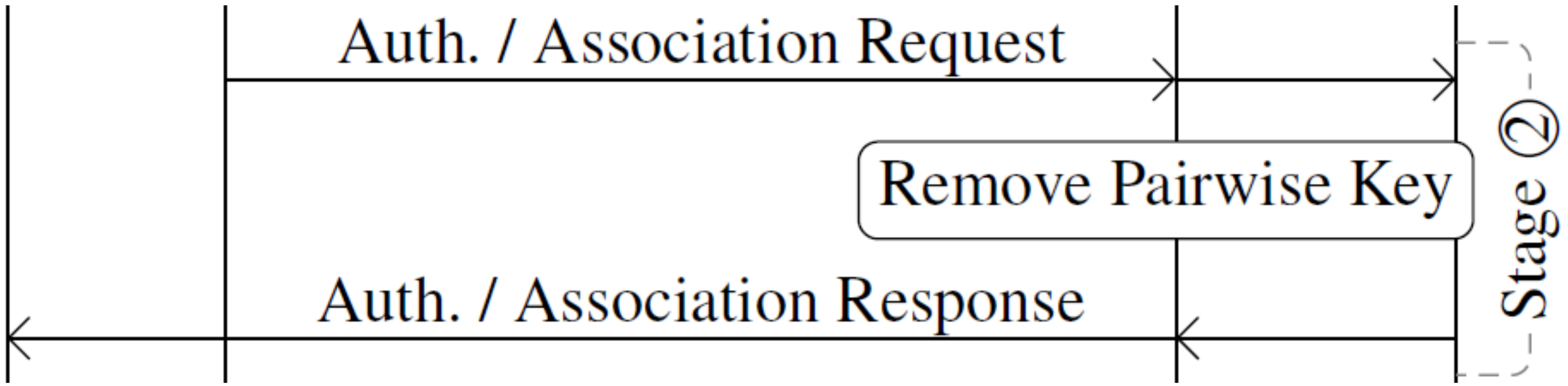


Attack 1: leaking frames

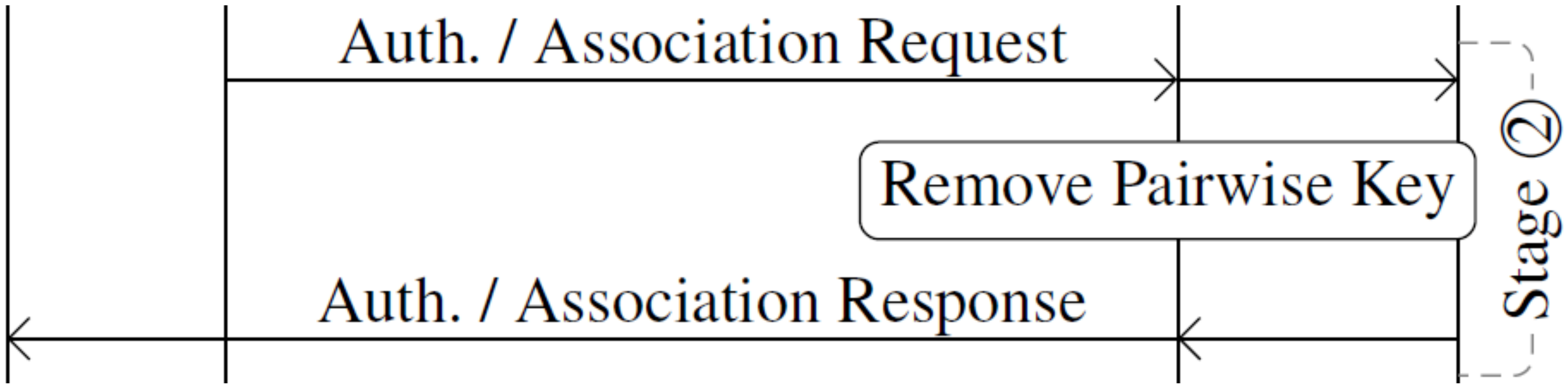


◎ **Connect to remove client's keys**

Attack 1: leaking frames



Attack 1: leaking frames



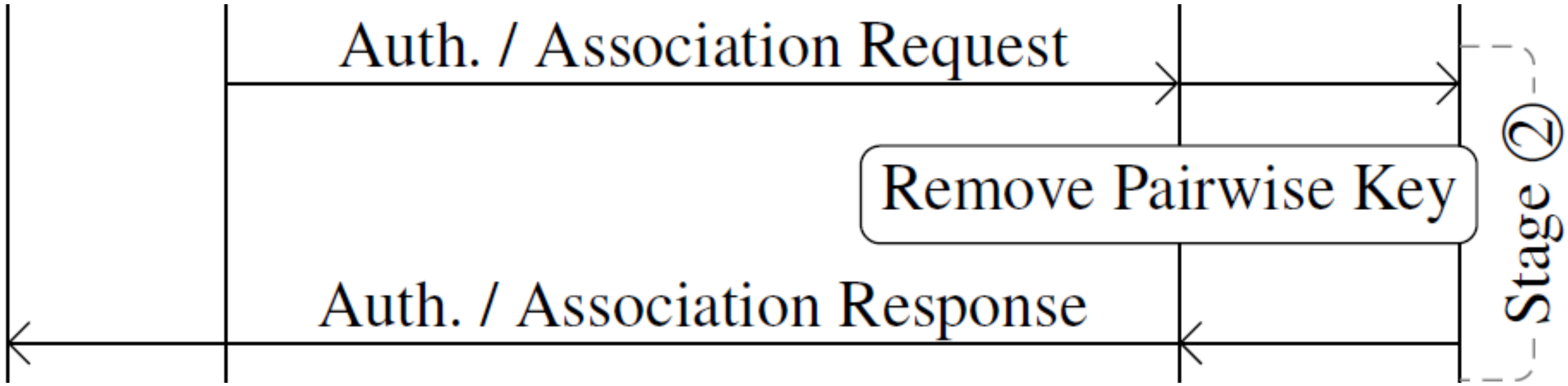
Auth. / Association Request

Remove Pairwise Key

Auth. / Association Response

Stage ②

Attack 1: leaking frames

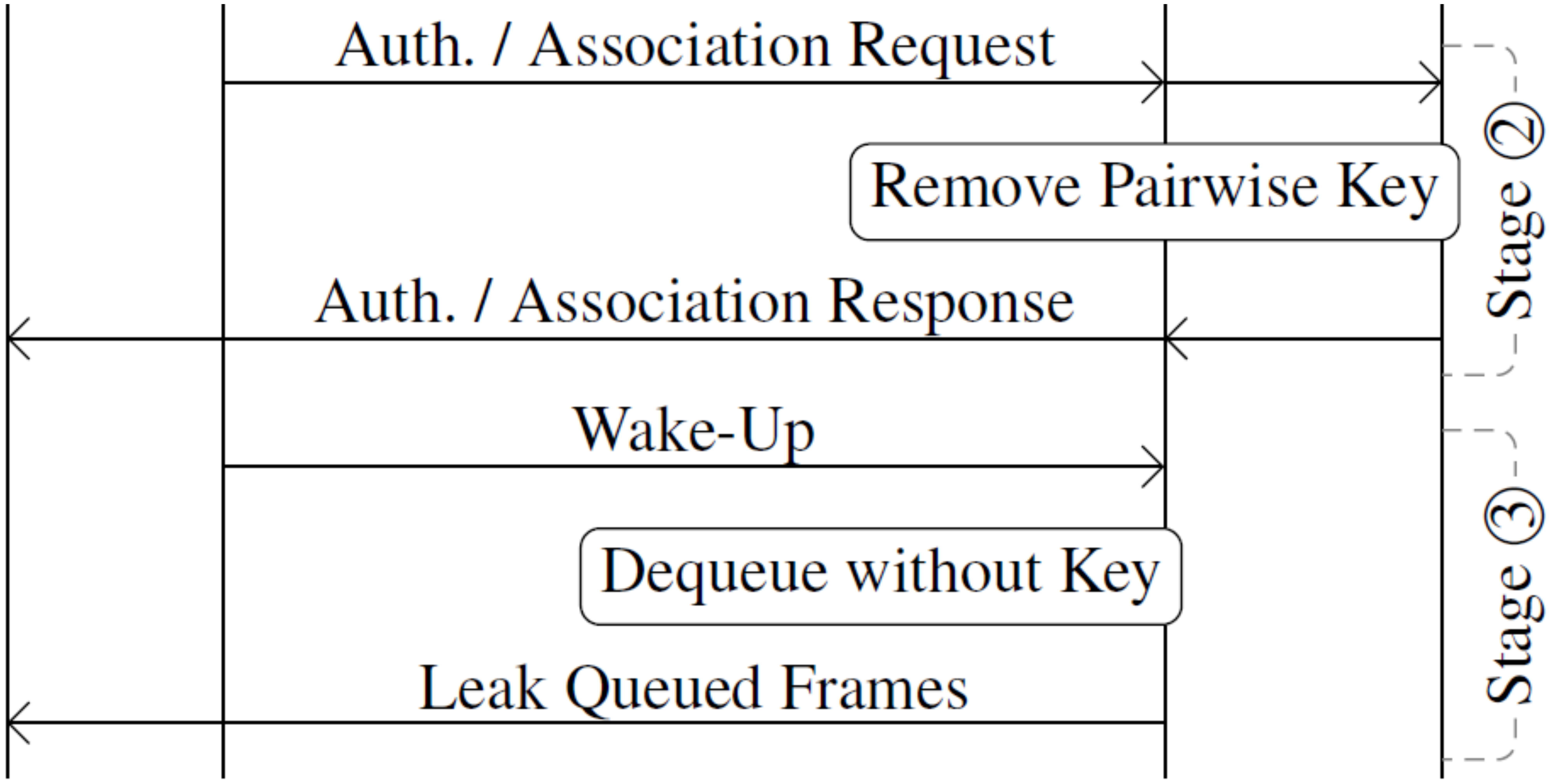


Auth. / Association Request

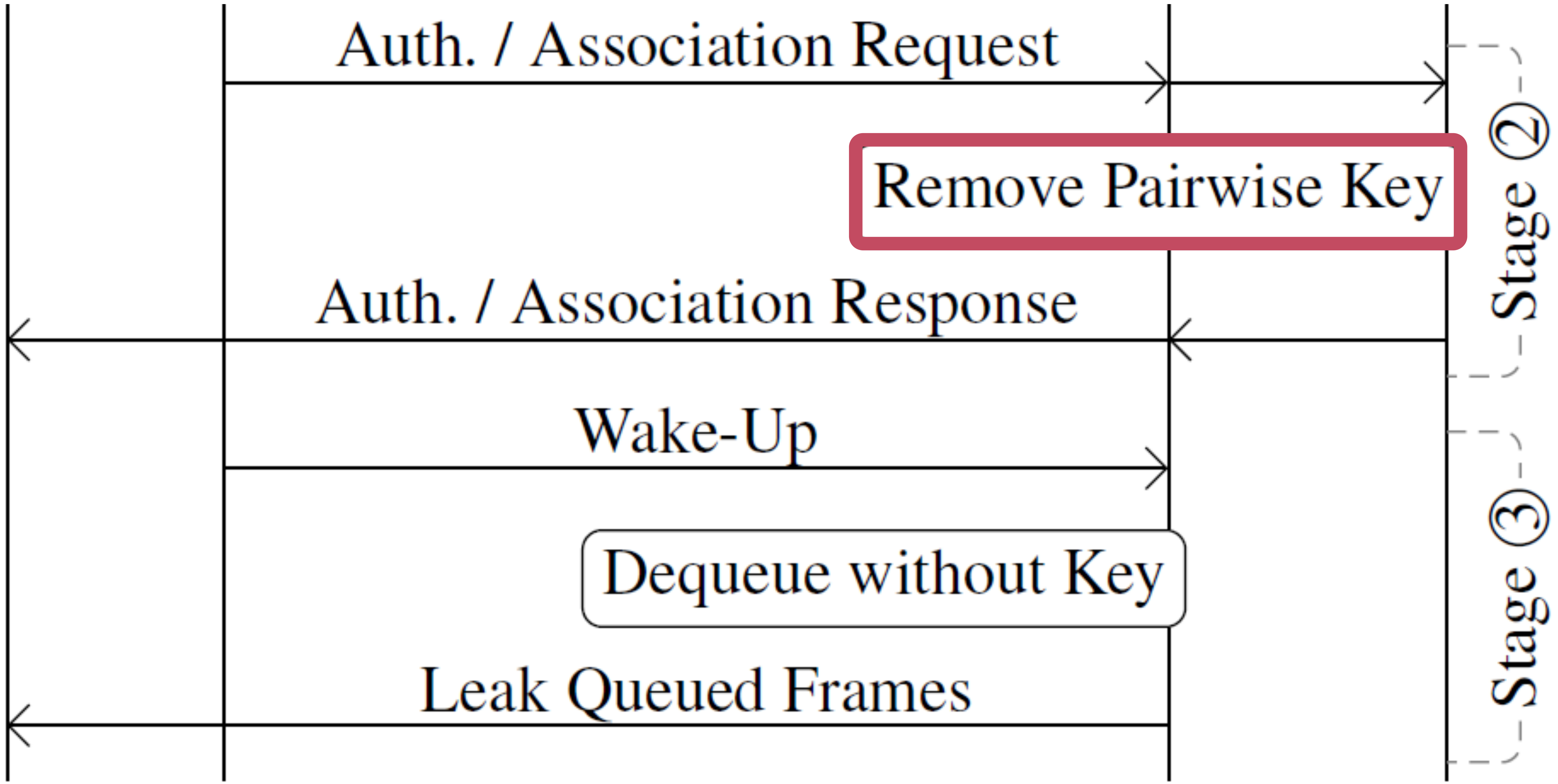
Remove Pairwise Key

Auth. / Association Response

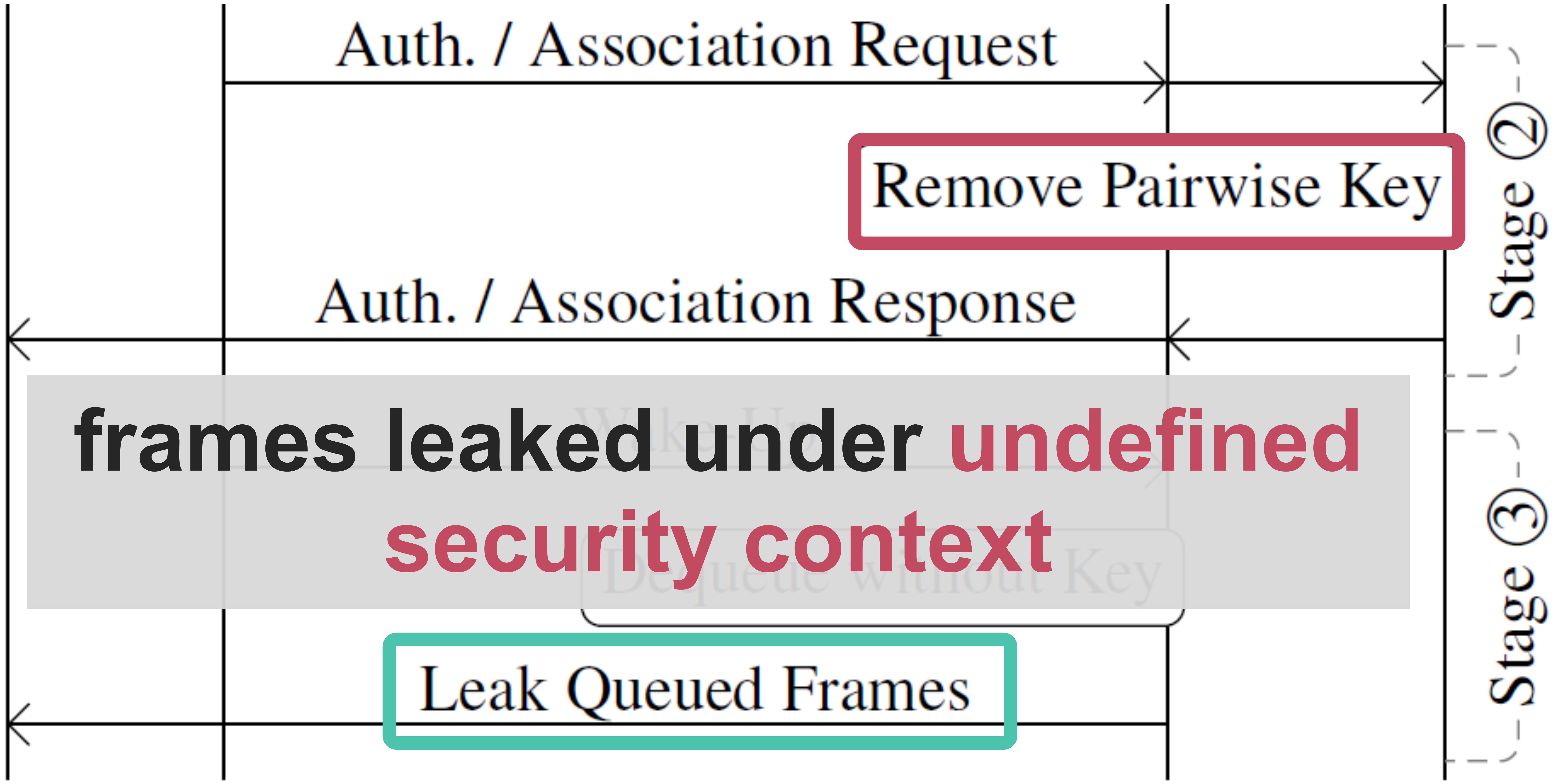
Attack 1: leaking frames



Attack 1: leaking frames



Attack 1: leaking frames



frames leaked under **undefined security context**

Undefined security context: FreeBSD example

How the frame is leaked depends on kernel version & driver:

Version	driver (vendor)	Leakage
13.0	run (Ralink)	Plaintext
13.1	run (Ralink)	WEP with all-zero key
13.1	rum (Ralink)	CCMP with group key
13.1	rtwn (Realtek)	CCMP with group key

Undefined security context: FreeBSD example

How the frame is leaked depends on kernel version & driver:

Version	driver (vendor)	Leakage
13.0	run (Ralink)	Plaintext
13.1	run (Ralink)	WEP with all-zero key
13.1	rum (Ralink)	CCMP with group key
13.1	rtwn (Realtek)	CCMP with group key

- Malicious insiders know the group key!
- Linux, NetBSD, open Atheros firmware also affected

Root cause

Standard isn't explicit on how to manage buffered frames

- Should drop buffered frames when refreshing/deleting keys

[CKM20]: A Formal Analysis of IEEE 802.11's WPA2 by C. Cremers, B. Kiesl, and N. Medinger (USENIX Security)

Root cause

Standard isn’t explicit on how to manage buffered frames

- Should drop buffered frames when refreshing/deleting keys

Lesson: include transmit queue in formal Wi-Fi models

- Because buffered frames are not yet encrypted (unlike TLS)
- [CKM20] modelled transmit queue but not key deletion!

[CKM20]: A Formal Analysis of IEEE 802.11’s WPA2 by C. Cremers, B. Kiesl, and N. Medinger (USENIX Security)

Finding 2: Bypassing Client Isolation

Attack 2: Bypassing Wi-Fi Client Isolation

Attack targets networks that use client isolation:

- Defense mechanism against malicious or compromised inside clients.
- Typically networks in large organizations, universities, public hotspots.

Wi-Free

eduroam

Hotspot 2.0

Attack 2: Bypassing Wi-Fi Client Isolation

Attack targets networks that use client isolation:

- Defense mechanism against malicious or compromised inside clients.
- Typically networks in large organizations, universities, public hotspots.

Wi-Free eduroam Hotspot 2.0

Attacker can connect to the network, but not communicate with others.

Attack 2: Bypassing Wi-Fi Client Isolation

Attack targets networks that use client isolation:

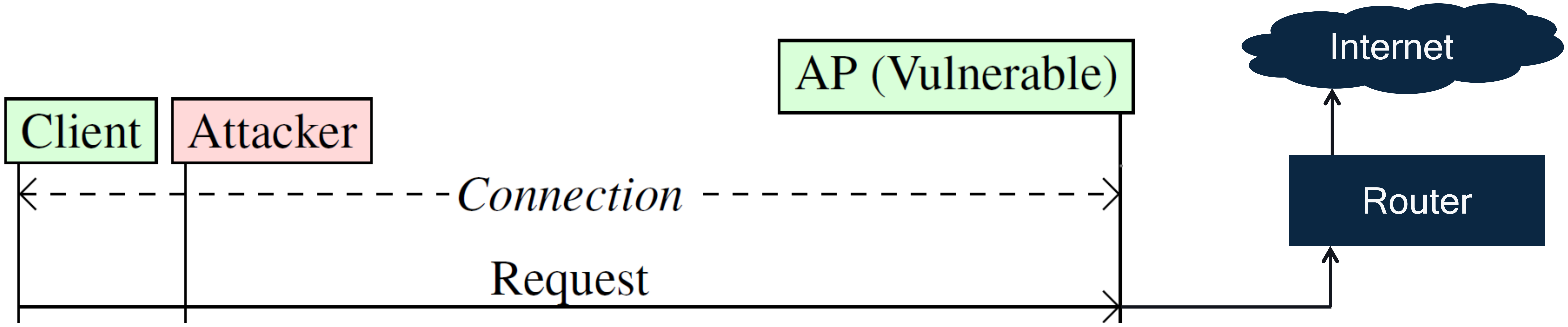
- Defense mechanism against malicious or compromised inside clients.
- Typically networks in large organizations, universities, public hotspots.

Wi-Free eduroam Hotspot 2.0

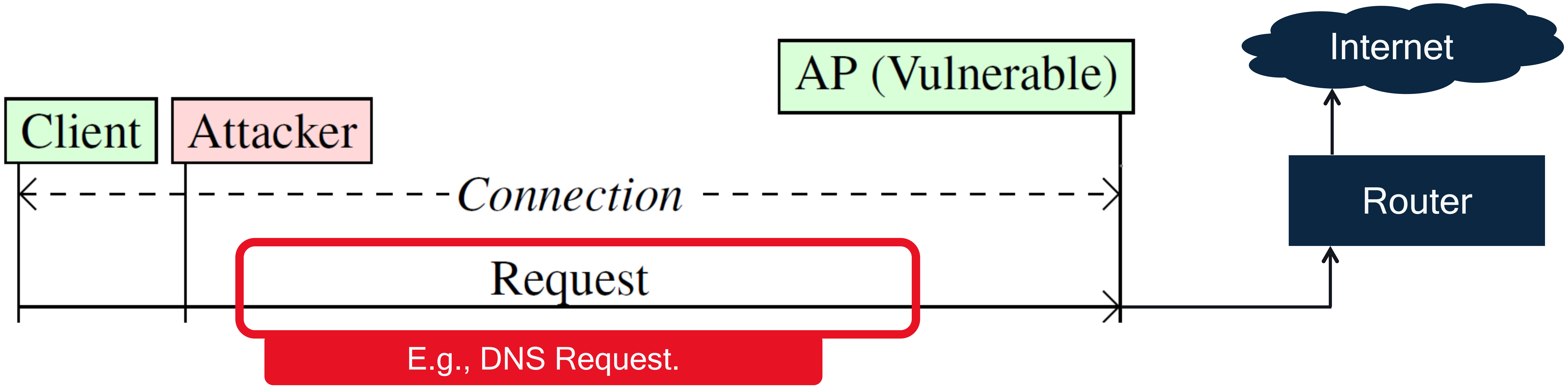
Attacker can connect to the network, but not communicate with others.

… unless we can manipulate the security context!

Attack 2: Bypassing Wi-Fi Client Isolation

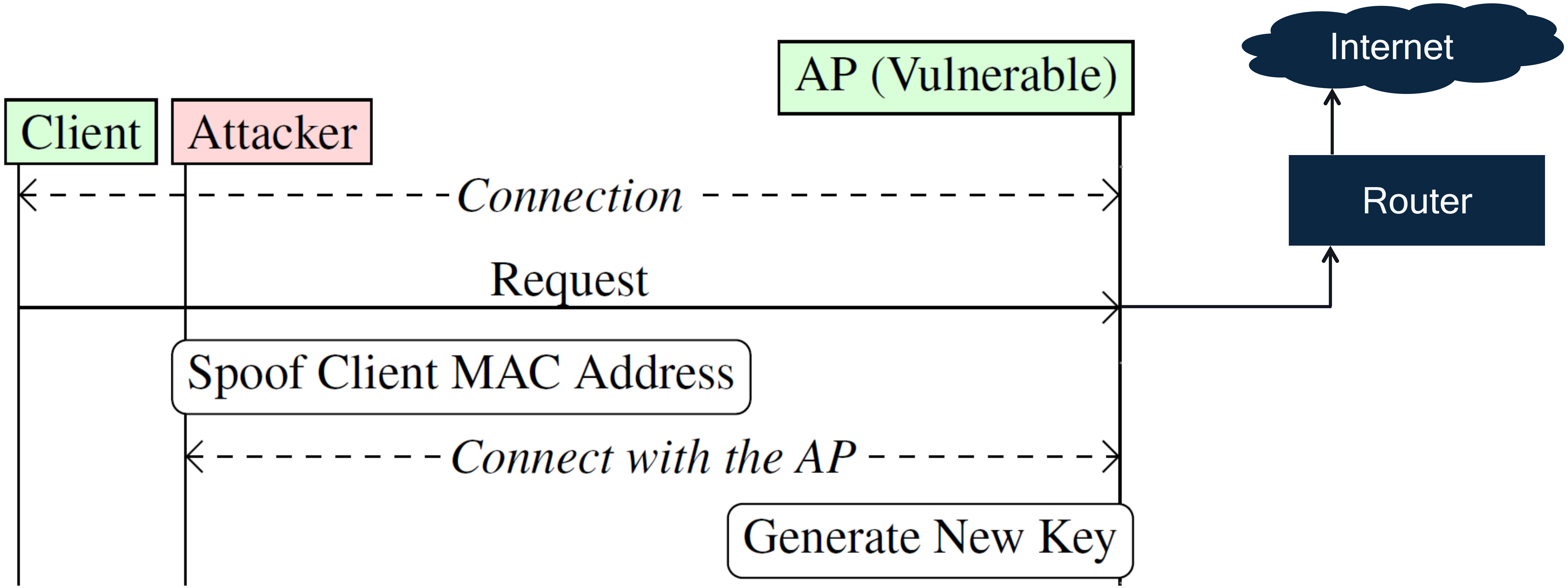


Attack 2: Bypassing Wi-Fi Client Isolation

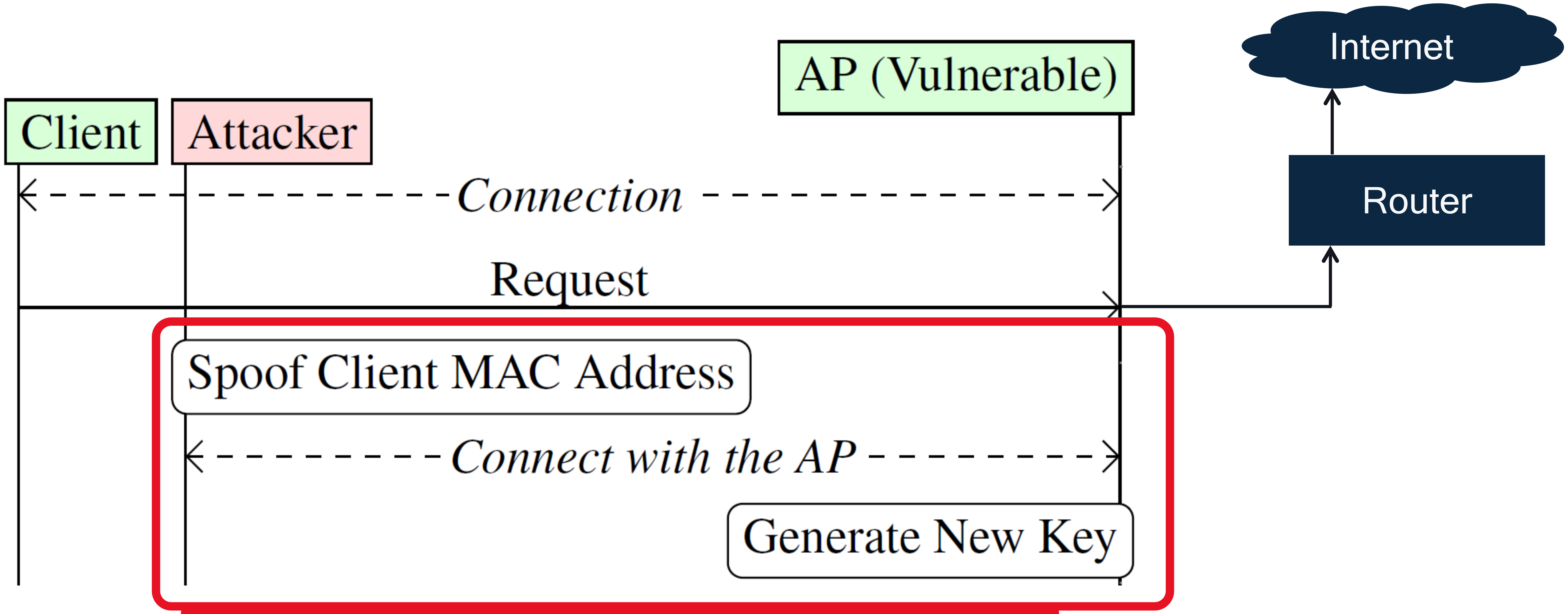


E.g., DNS Request.

Attack 2: Bypassing Wi-Fi Client Isolation

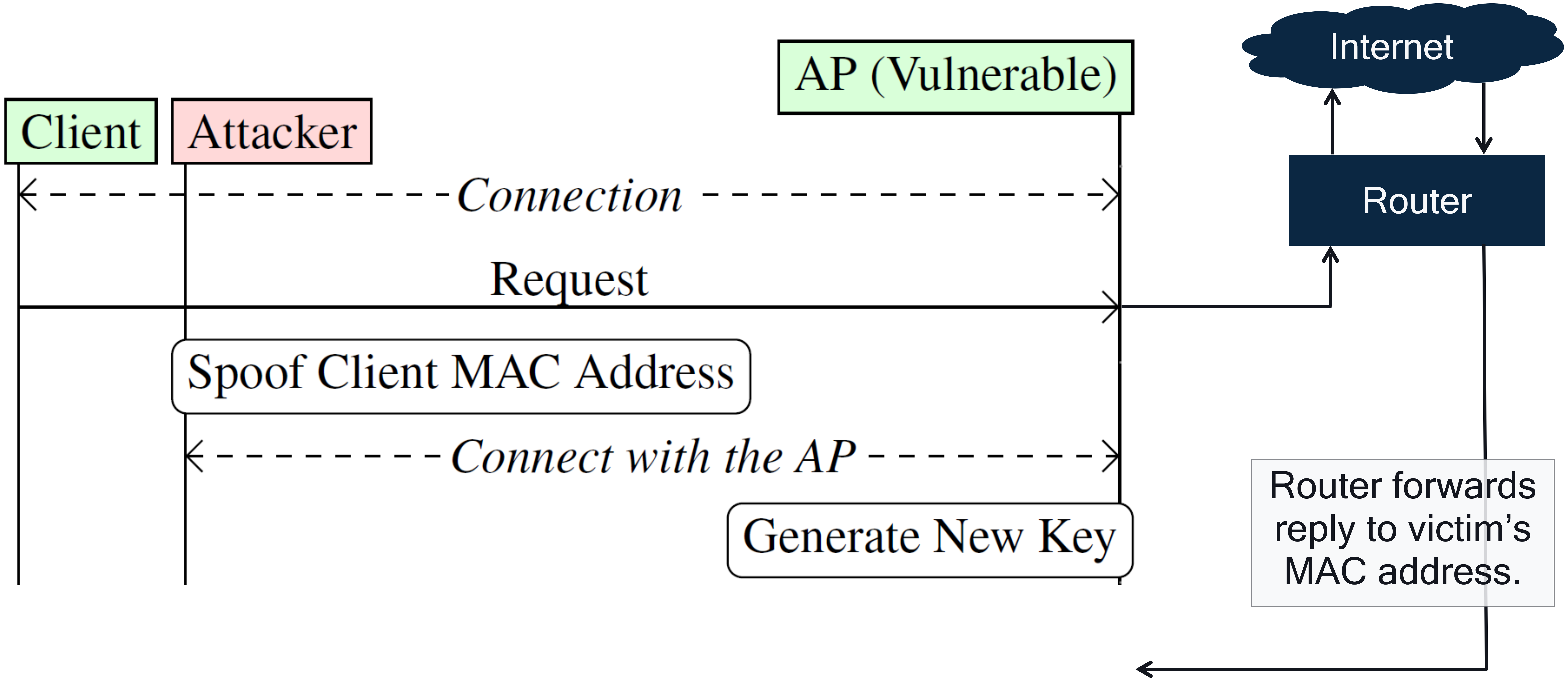


Attack 2: Bypassing Wi-Fi Client Isolation



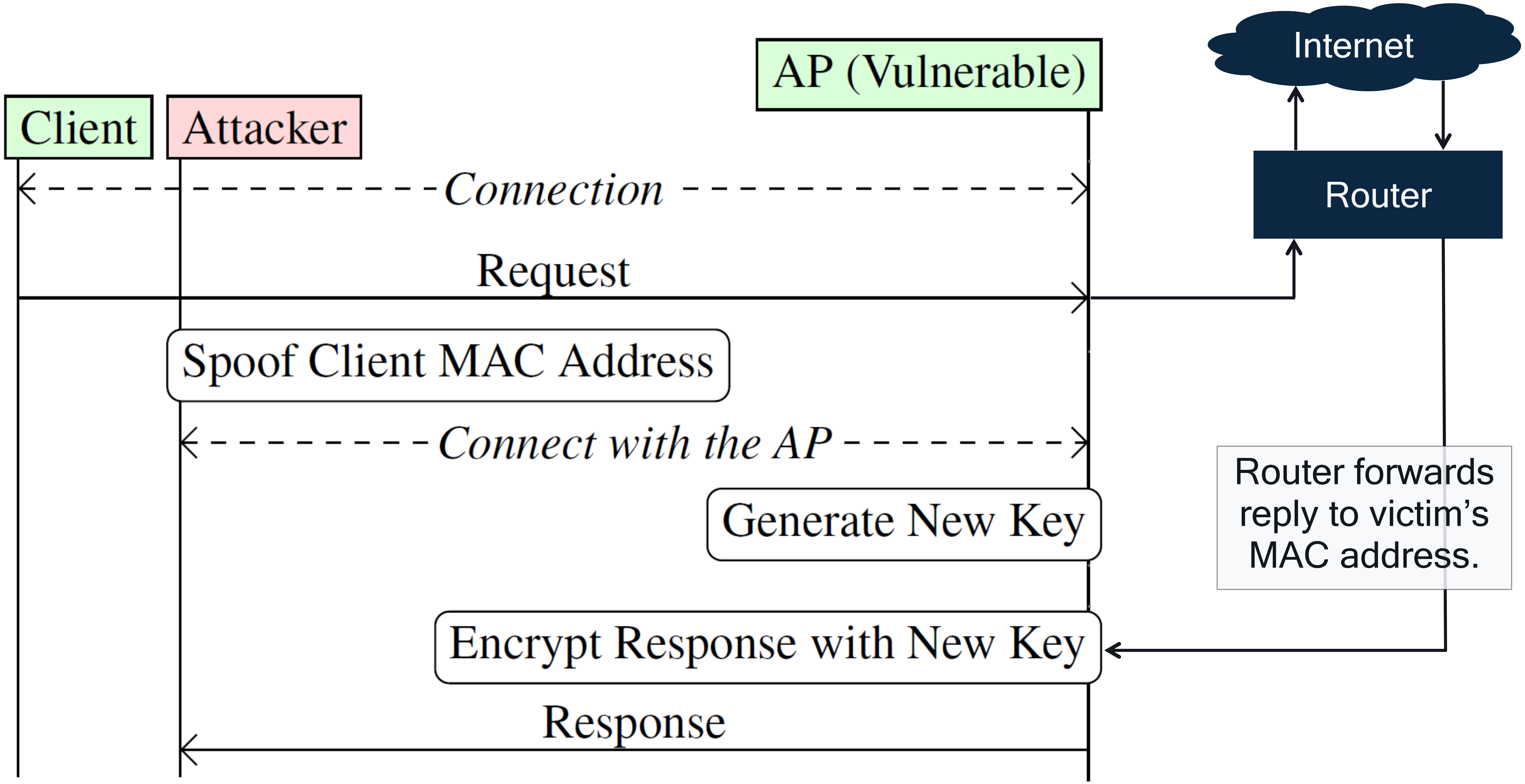
Adversary now owns the MAC and Encryption Key.

Attack 2: Bypassing Wi-Fi Client Isolation



Router forwards reply to victim’s MAC address.

Attack 2: Bypassing Wi-Fi Client Isolation



Router forwards reply to victim's MAC address.

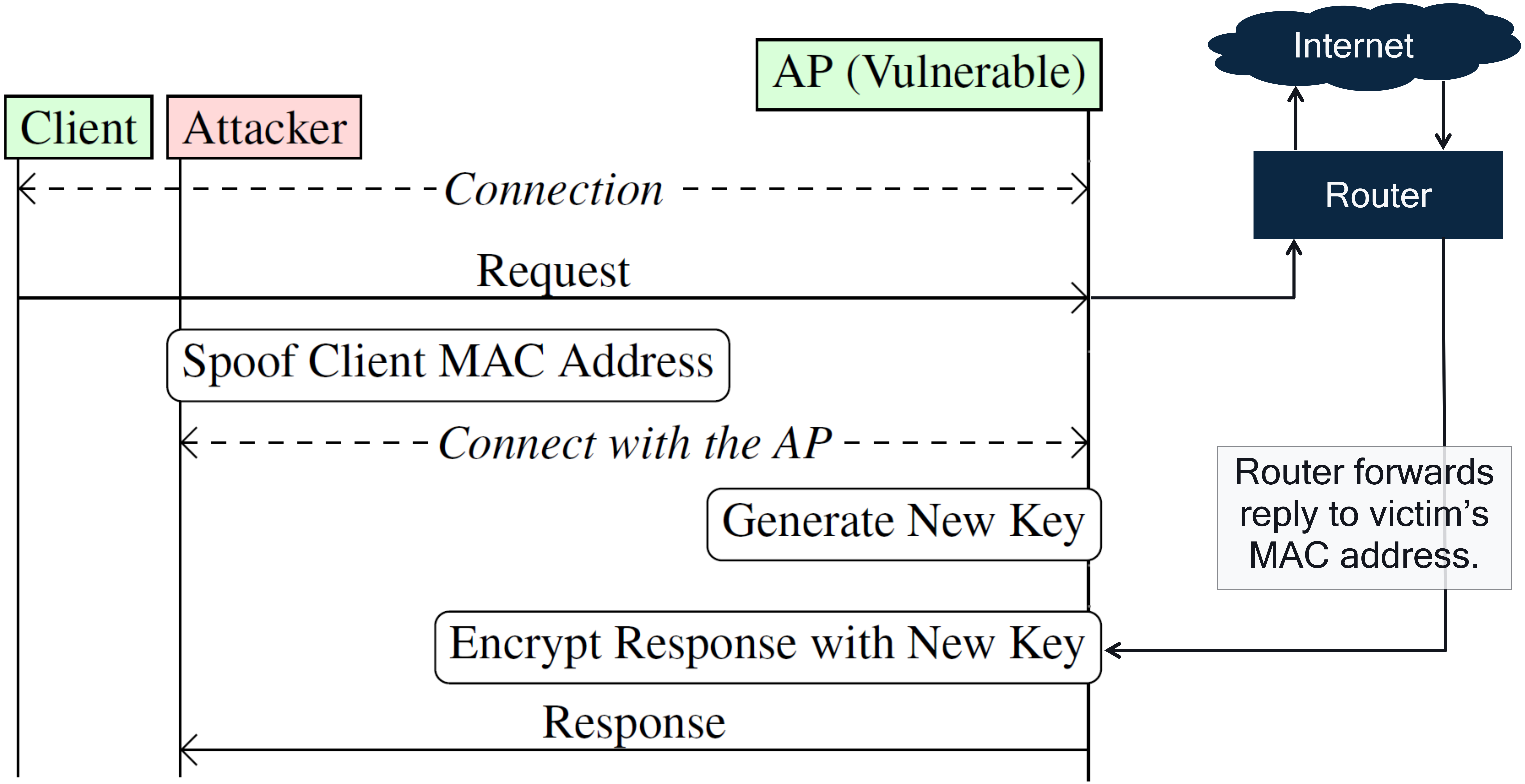
Experiments: home APs

Experiments: home APs

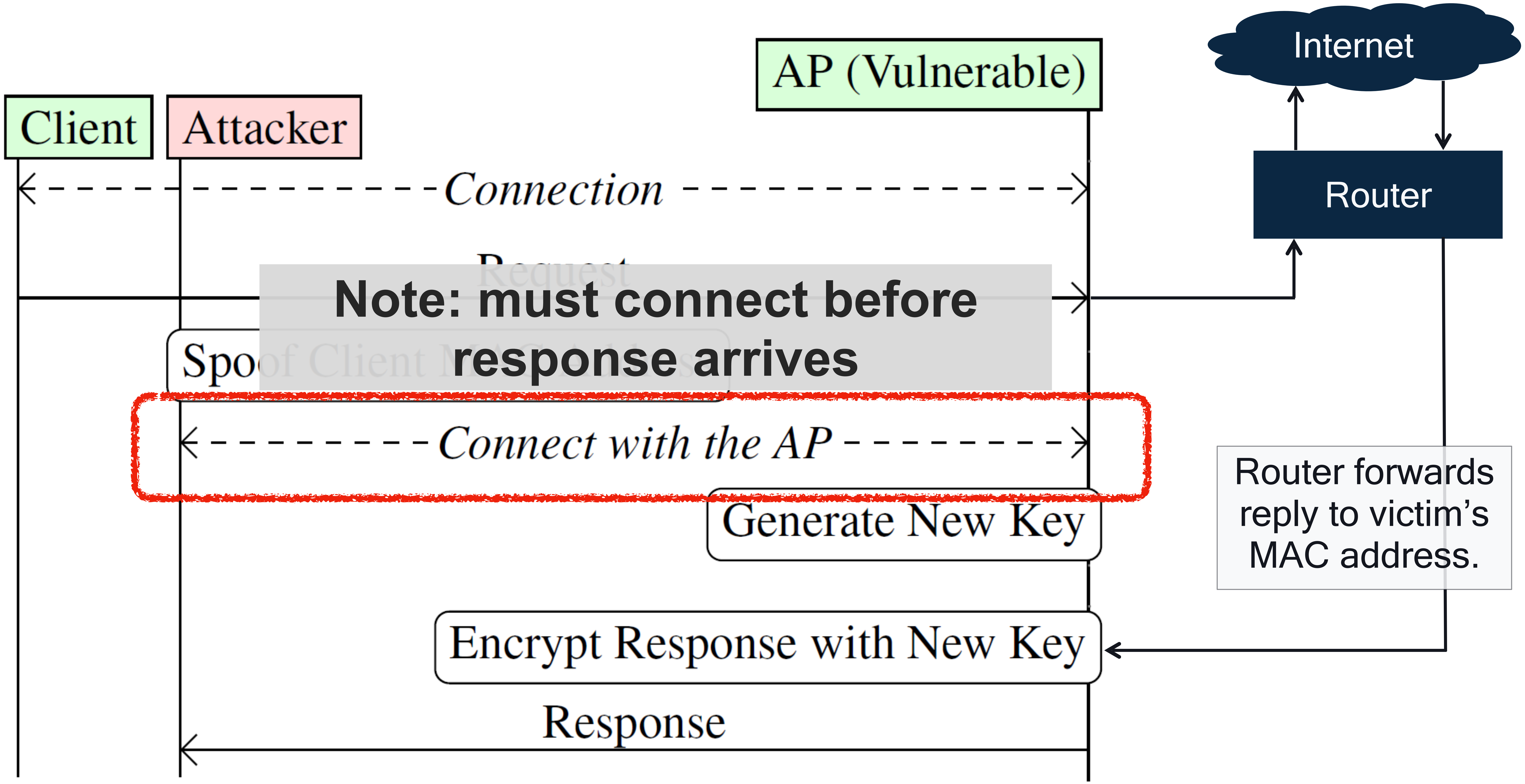
All tested professional & home APs were vulnerable

→ **Design flaw** in Wi-Fi client isolation!

Attack 2: Bypassing Wi-Fi Client Isolation



Attack 2: Bypassing Wi-Fi Client Isolation



Note: must connect before response arrives

Router forwards reply to victim's MAC address.

Attack 2: Bypassing Wi-Fi Client Isolation

Think of it as a **fast security context override**.

- Requires the attacker to reconnect within certain time restrictions.
- Timing restrictions no concern within transatlantic connections (UDP ~ 70 ms), reasonable within European connections (UDP ~13 ms).
- Protocols such as TCP retransmit when not acknowledged, thus trivial to intercept.

Attack 2: Bypassing Wi-Fi Client Isolation

Think of it as a **fast security context override**.

- Requires the attacker to reconnect within certain time restrictions.
- Timing restrictions no concern within transatlantic connections (UDP ~ 70 ms), reasonable within European connections (UDP ~13 ms).
- Protocols such as TCP retransmit when not acknowledged, thus trivial to intercept.

Adversary can spoof MAC address of a server or gateway in the LAN.

Attack 2: Bypassing Wi-Fi Client Isolation

Why?

Client identities are not bound to each other:

- IEEE 802.1X Identity (username), and
- IP/MAC Addresses.

No concept of ‘protected ownership of a MAC address’ (as is the case in IEEE 802 LANs).

Thus, an adversary can spoof the client’s identity on other layers.

Attack 2: Bypassing Wi-Fi Client Isolation

Why?

Client identities are not bound to each other:

- IEEE 802.1X Identity (username), and
- IP/MAC Addresses.

No concept of ‘protected ownership of a MAC address’ (as is the case in IEEE 802 LANs).

Thus, an adversary can spoof the client’s identity on other layers.

Design shortcomings/limitations in the standard, network.

Attack 2: Bypassing Wi-Fi Client Isolation

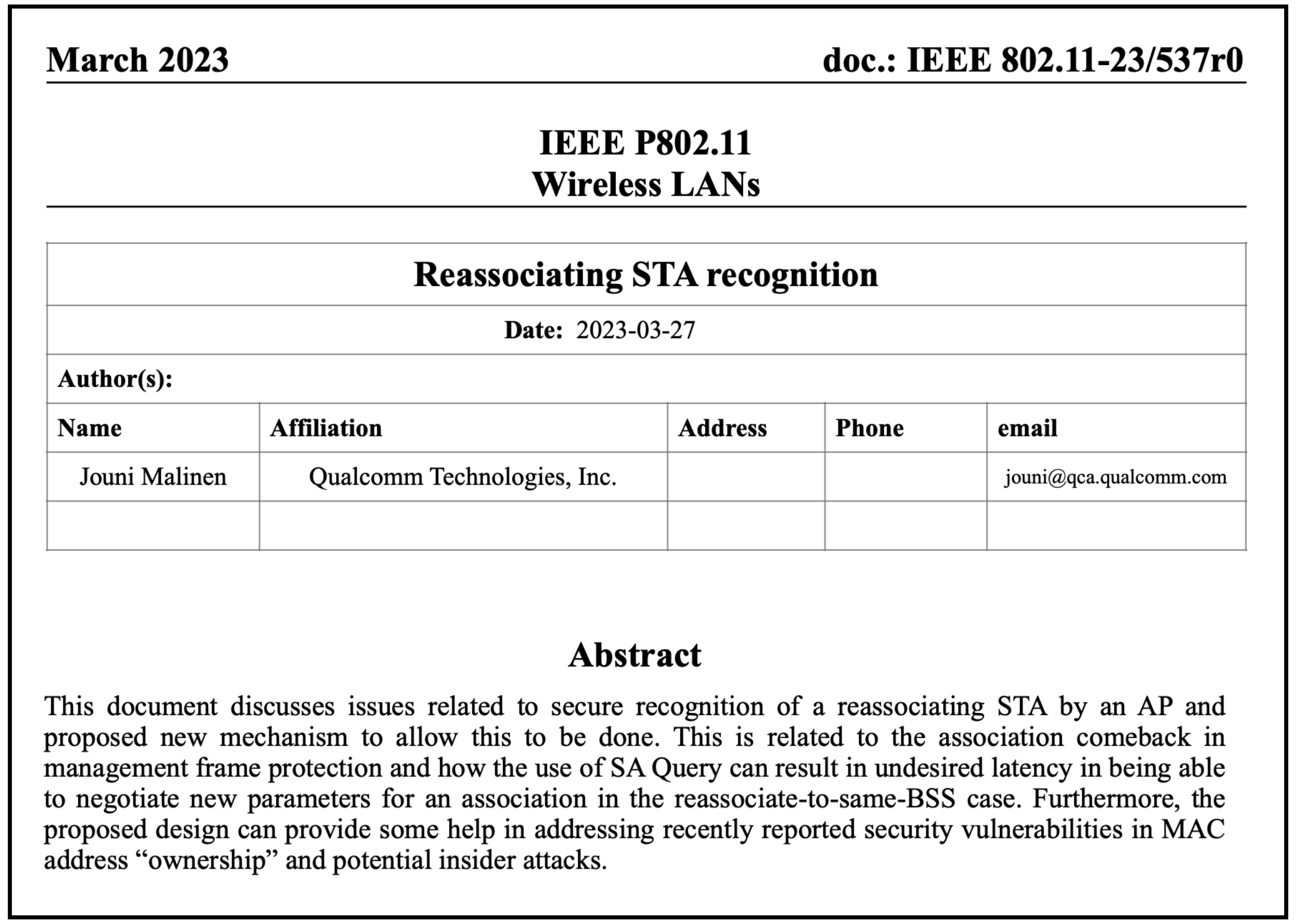
- This is not a simple (or difficult) code fix for anyone.
- Needs to be addressed within multiple network components, beyond an access point.

Solutions? Probably not realistic, practical, or sufficient:

- Reject recently-used MAC addresses (e.g., a ten second delay if client isolation is configured).
- Network configurations to use separate (un)trusted clients (e.g., different SSIDs, usage of VLANs).
- Require connection establishments to use a cached key if recently-used MAC address.

Summary

- Standard is vague and requires explicit elaboration on managing buffered frames
    - Can leak frames under different security context
    - Important to model/define transmit queues
- Can bypass client isolation
    - All devices vulnerable -> design flaw
    - Hard to fully prevent
- Some DoS attacks also possible (paper has details)



March 2023 doc.: IEEE 802.11-23/537r0

IEEE P802.11
Wireless LANs

Reassociating STA recognition

Date: 2023-03-27

Author(s):

Name	Affiliation	Address	Phone	email
Jouni Malinen	Qualcomm Technologies, Inc.			jouni@qca.qualcomm.com

Abstract

This document discusses issues related to secure recognition of a reassociating STA by an AP and proposed new mechanism to allow this to be done. This is related to the association comeback in management frame protection and how the use of SA Query can result in undesired latency in being able to negotiate new parameters for an association in the reassociate-to-same-BSS case. Furthermore, the proposed design can provide some help in addressing recently reported security vulnerabilities in MAC address "ownership" and potential insider attacks.

https://mentor.ieee.org/802.11/dcn/23/11-23-0537-00-000m-

GitHub

https://github.com/vanhoefm/macstealer

https://github.com/domienschepers/wifi-framing

CVE-2022-47522

Thank you!