Abusing the Windows WiFi native API to create a covert channel

Andrés Blanco
Ezequiel Gutesman

Outline

- Covert Channels
- Attack Vectors and Scenarios
- IEEE 802.11 Fundamentals
- Covert Channel Design
- Implementation
- Demo
- Future Work and Enhancements

What's a covert channel?

“... any communication channel that can be exploited by a process to transfer information in a manner that violates the system's security policy.”

Department of Defense Trusted Computer System Evaluation Criteria

What's a covert channel?

Hiding information inside "safe" network packets could be used to bypass network security protections.

(e.g., HTTP proxies, Firewalls, IDS/IPS, etc.)

What's a covert channel?

Why should we try to pass through the security measures, when we can fly over it.

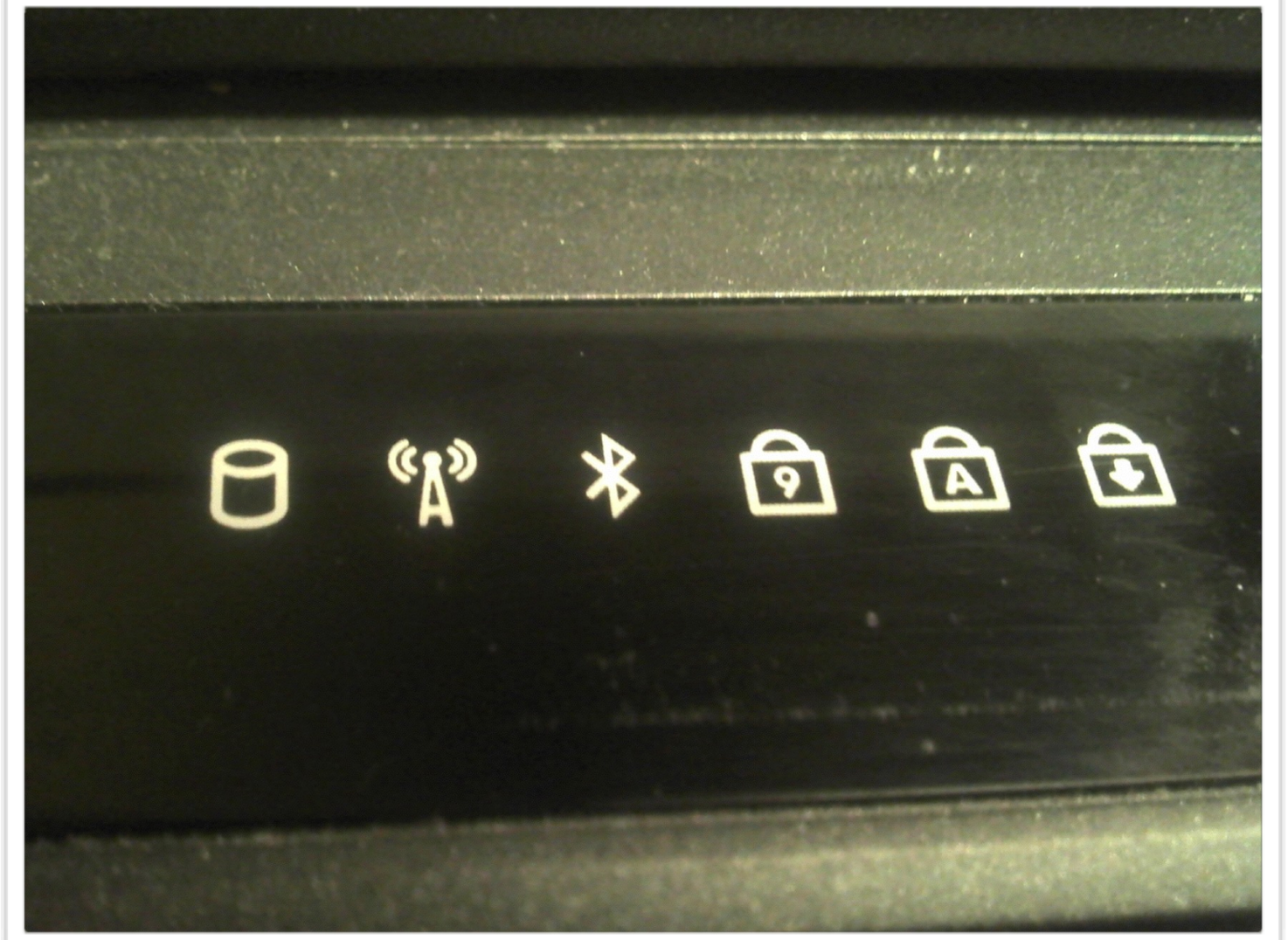
Network Boundaries

Like a castle

Network Boundaries

The old days

Network Boundaries



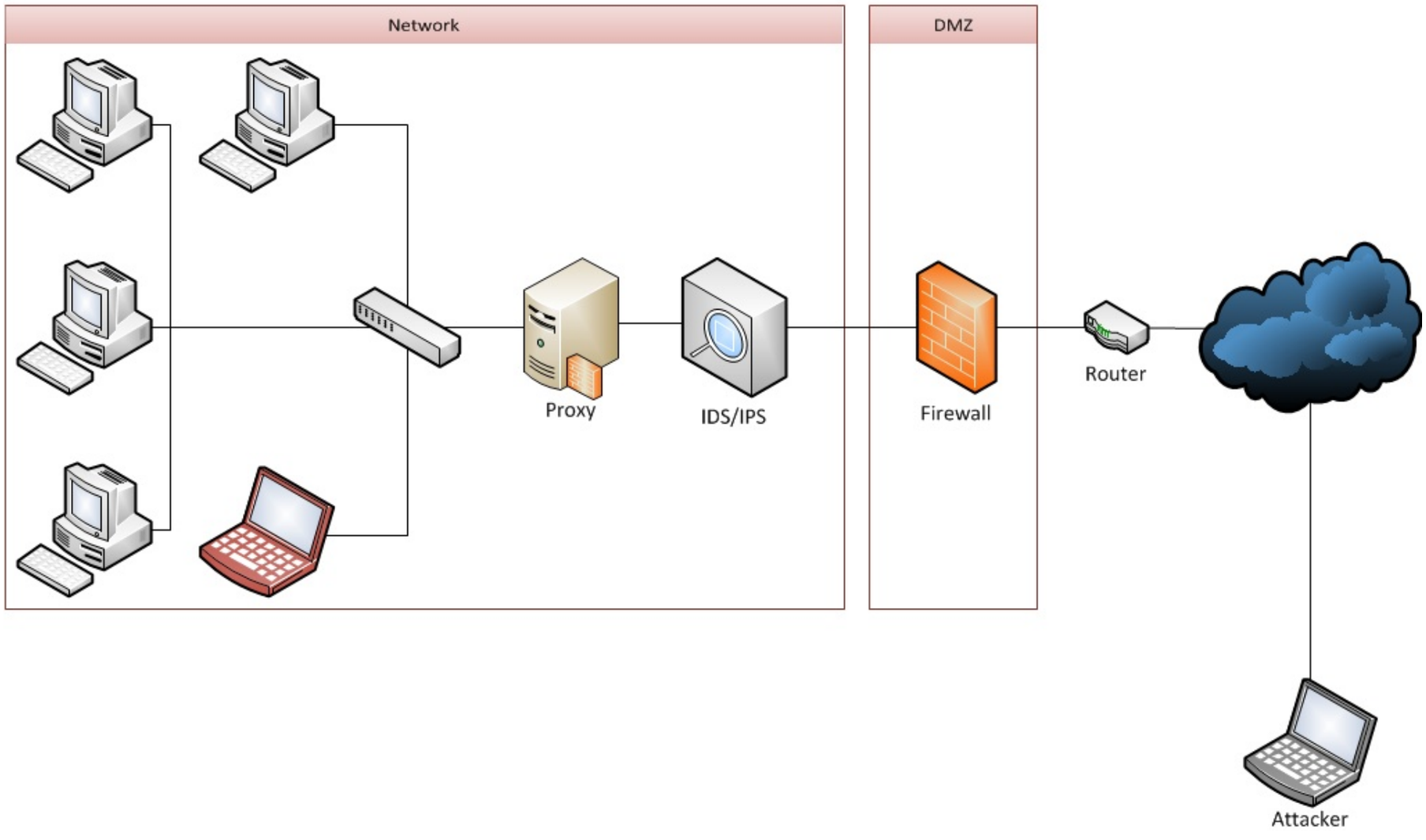
Nowadays

Network Boundaries

From secure to unsecured

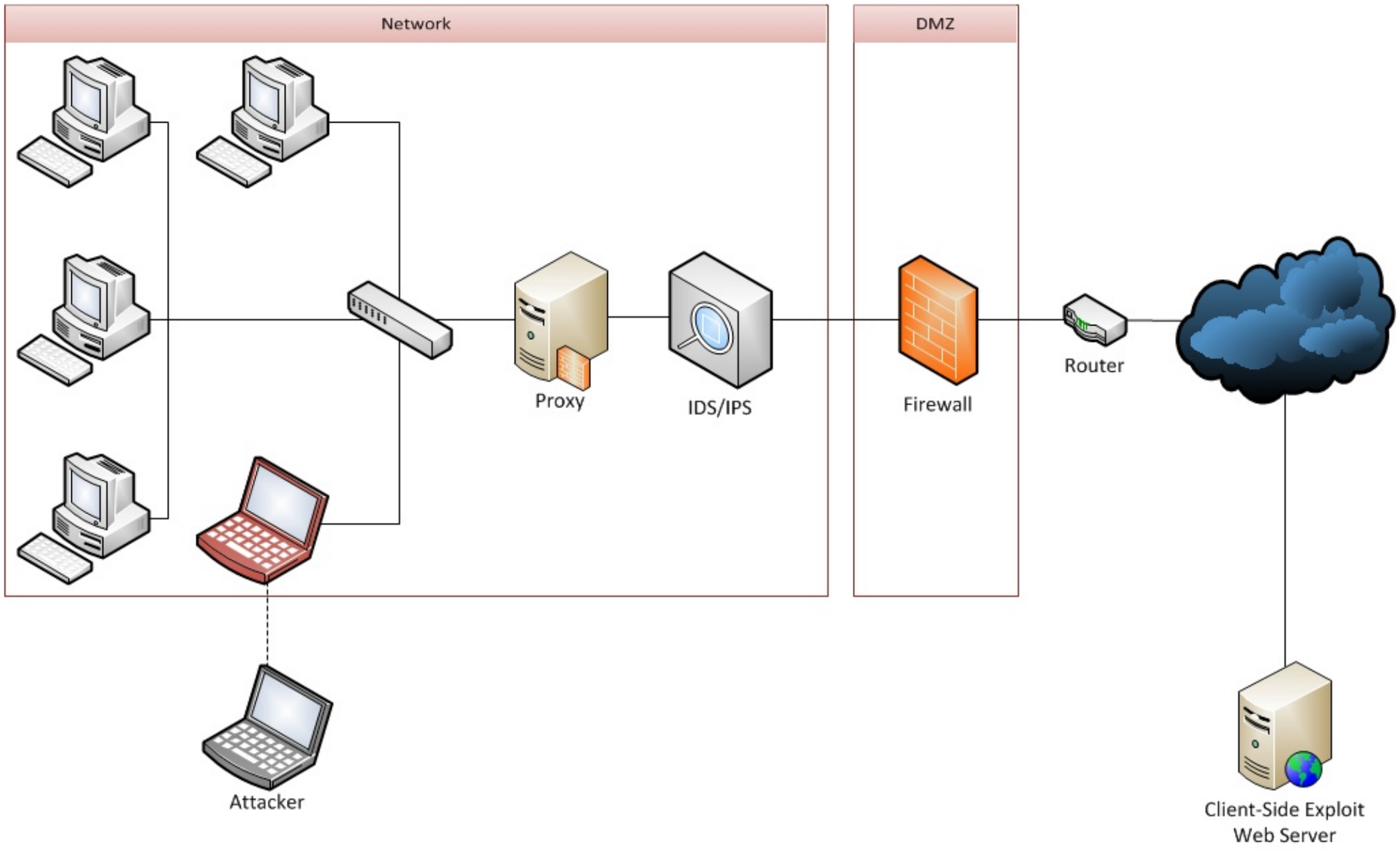
Attack Vectors & Scenarios

Escaping the hard way



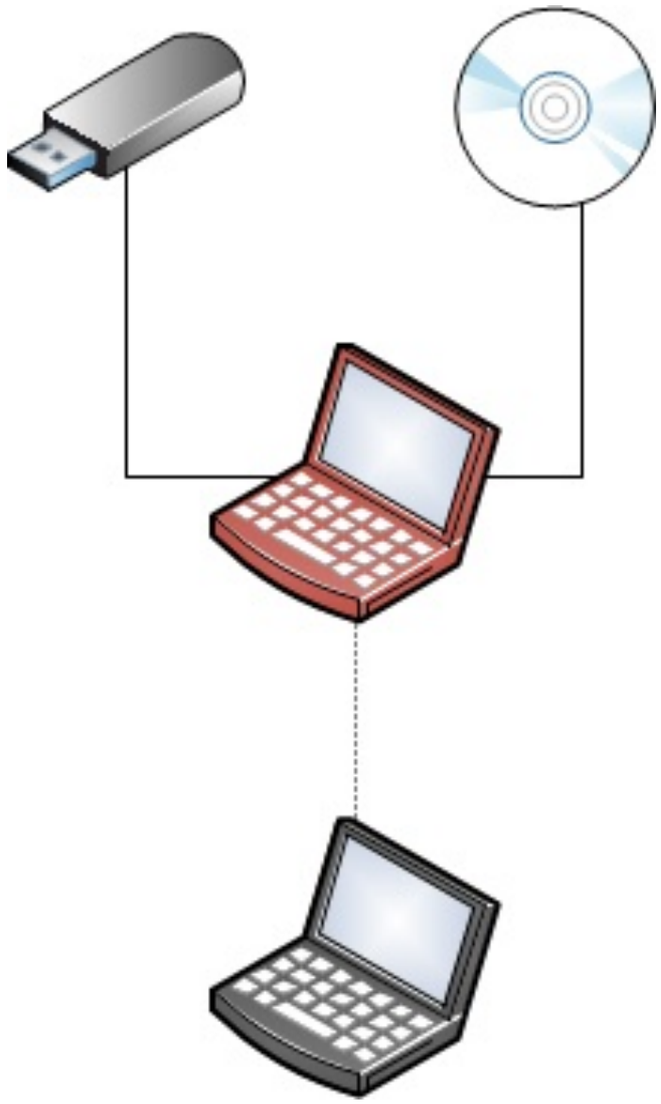
Attack Vectors & Scenarios

Jumping the fence



Attack Vectors & Scenarios

Attacking hosts with no connectivity



Prior Art

- MS Windows Soft AP
- Vendor-specific Soft AP

Comparison with MS Windows SoftAP

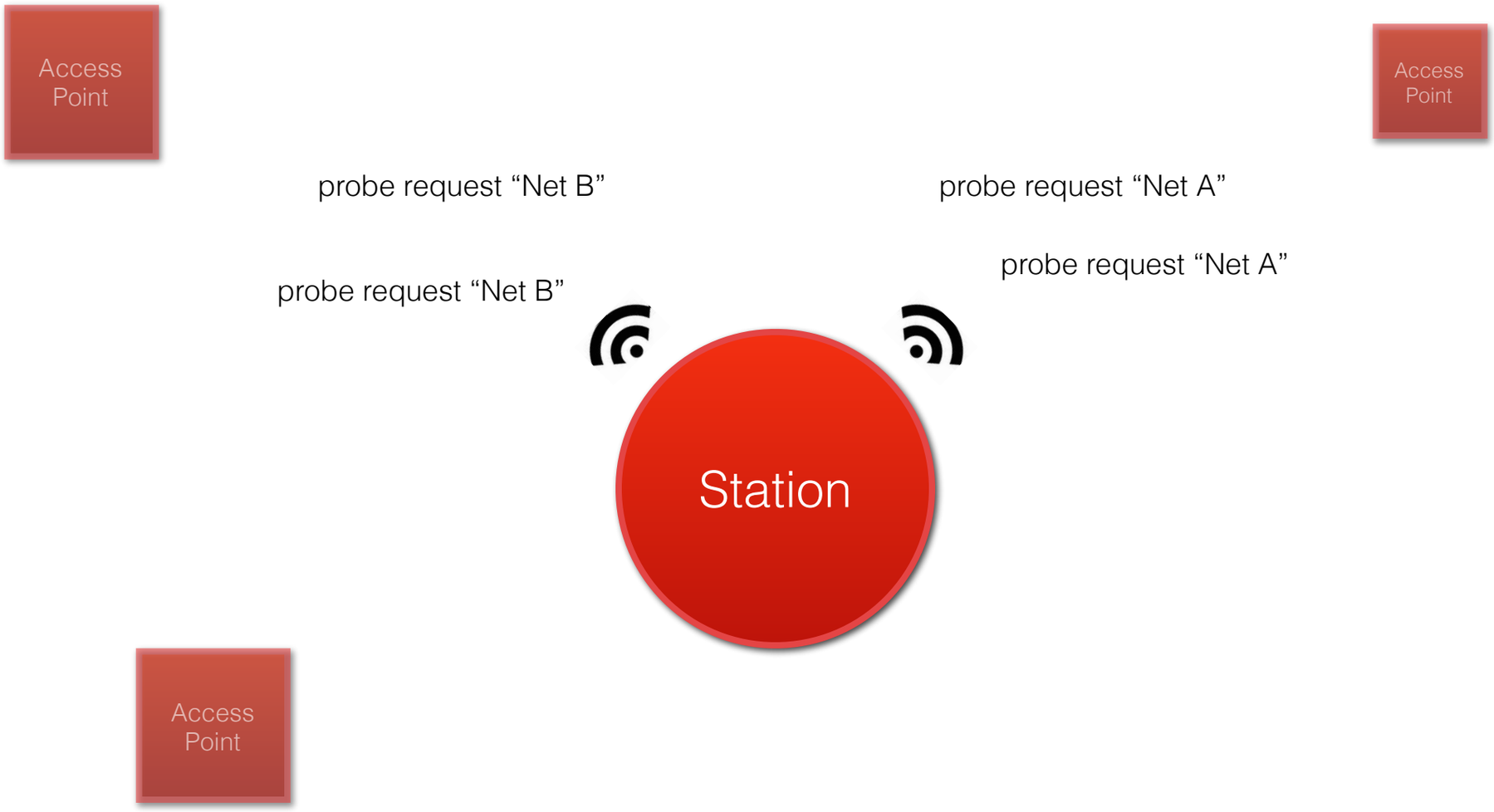
Windows SoftAP	WiFi Native API Covert Channel
Supported from Windows 7 and Windows Server 2008 R2 or later	Supported from Windows Vista or later
Needs administrator privileges	Doesn't need administrator privileges
Good bandwidth	Limited bandwidth
Not supported on every Windows driver To receive the Windows 7 logo, a wireless driver must implement the wireless Hosted feature.	Should work with any driver that works on Windows
User can notice the SoftAP is running	Hidden from user

IEEE 802.11 Fundamentals

AP Announcement

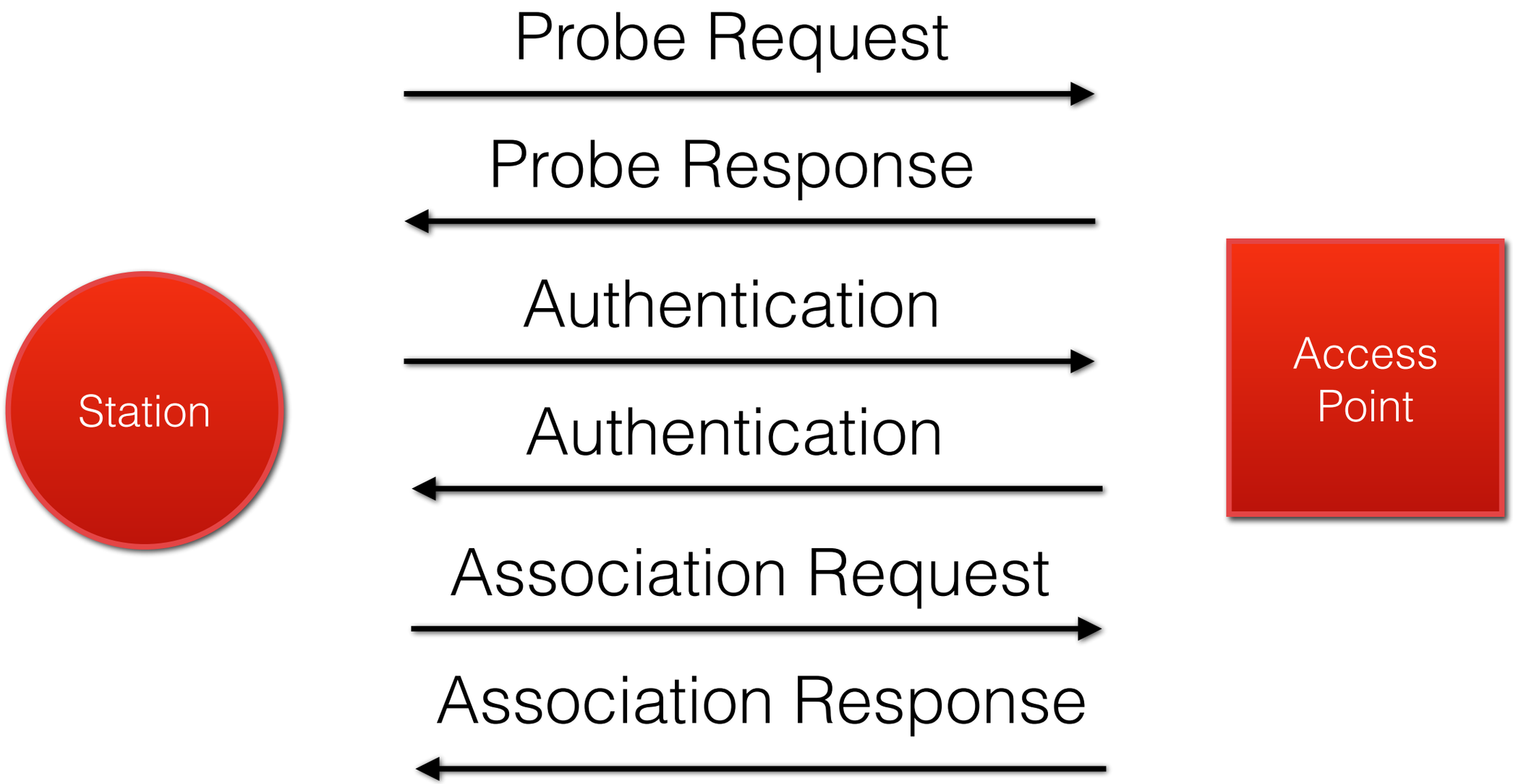
IEEE 802.11 Fundamentals

Active Scan for networks



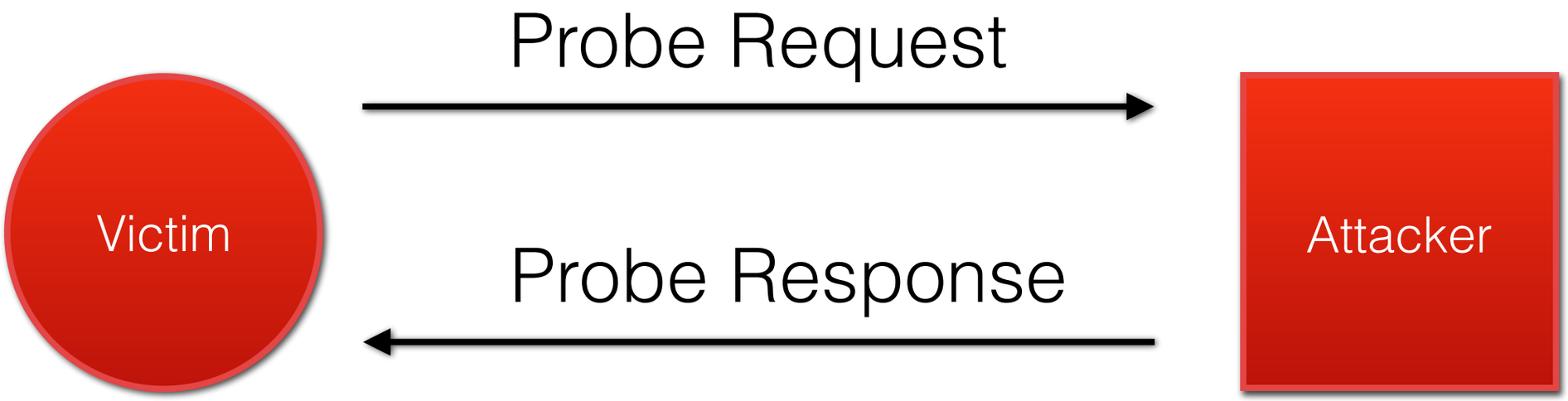
IEEE 802.11Fundamentals

Joining a network



Covert Channel Design

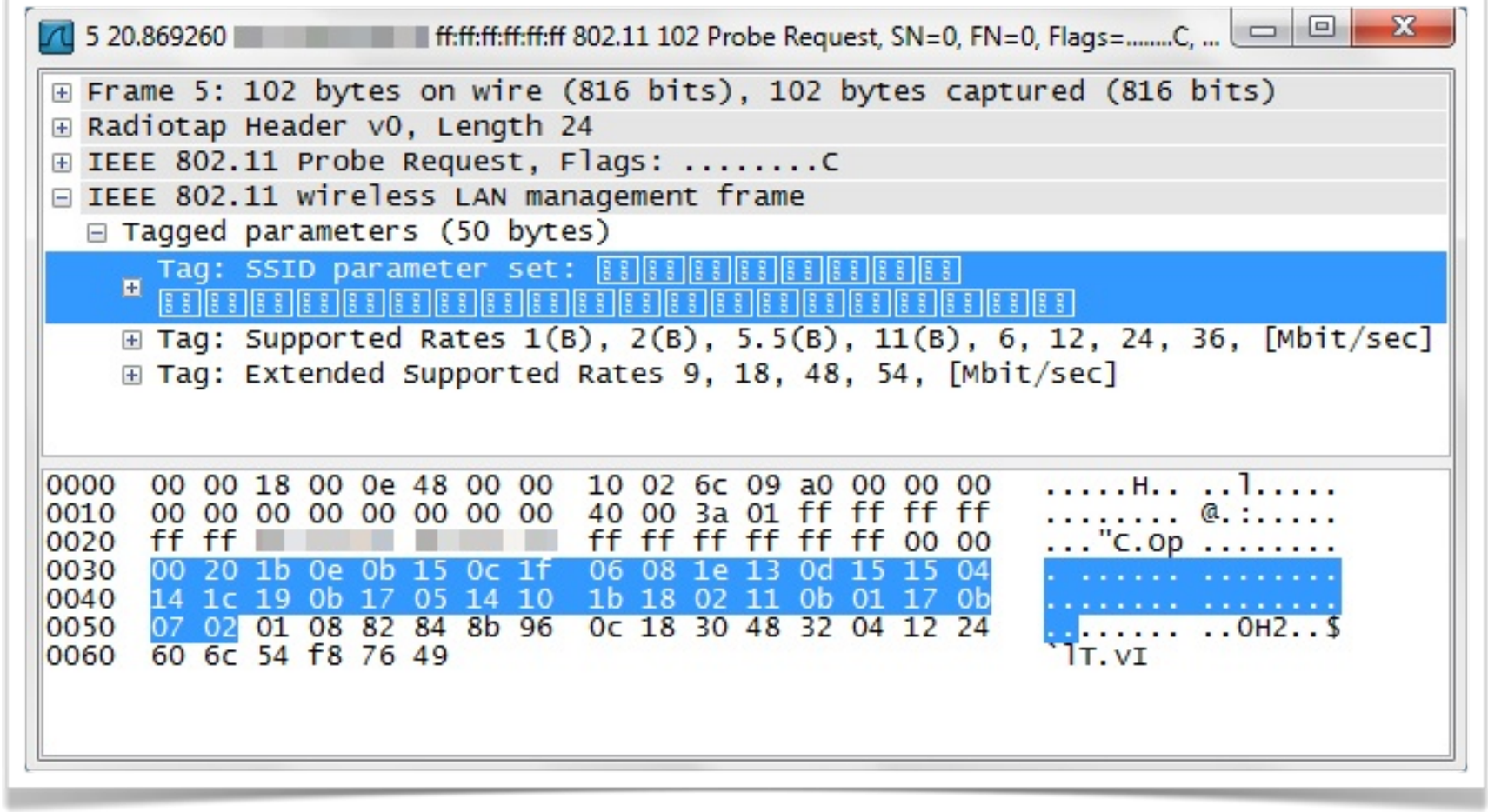
Hiding ourselves



Ref: Attacking Automatic Wireless Network Selection (http://www.theta44.org/karma/aawns.pdf)

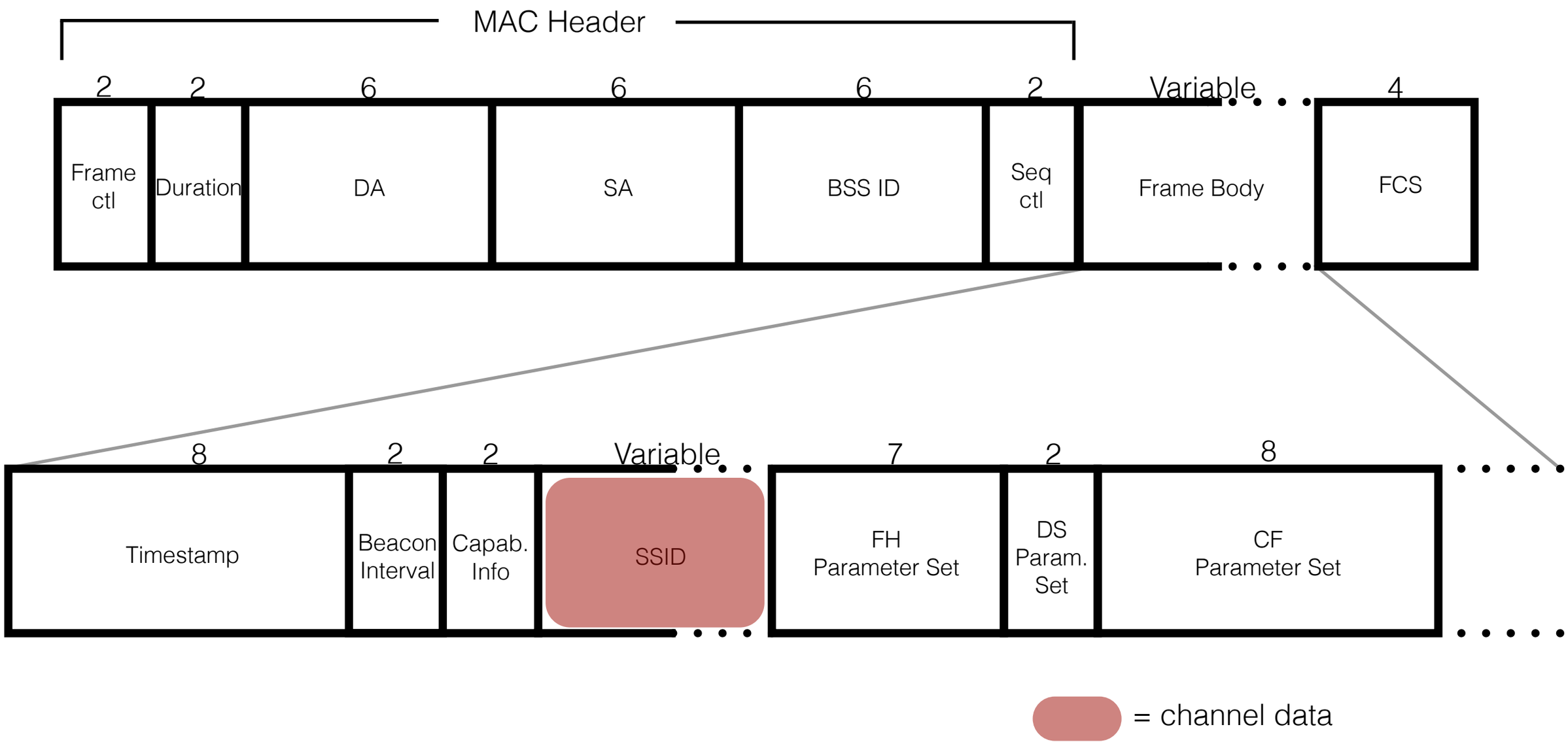
Covert Channel Design

Hiding ourselves



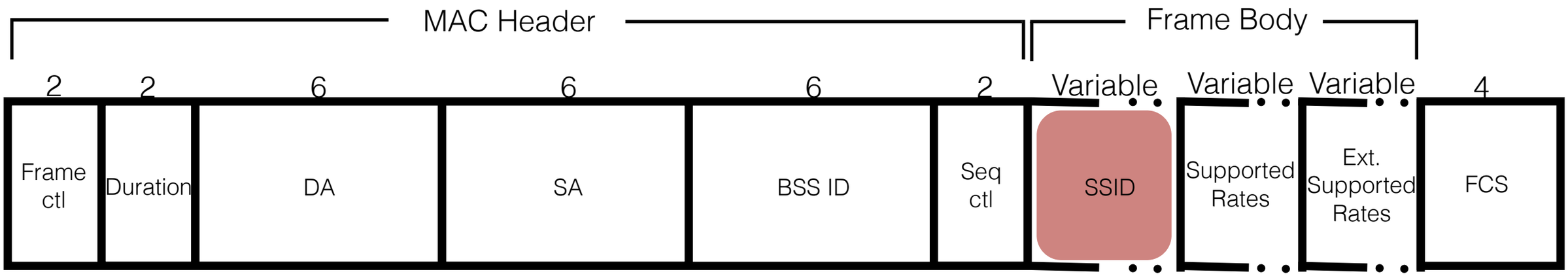
Covert Channel Design

Beacon Frames



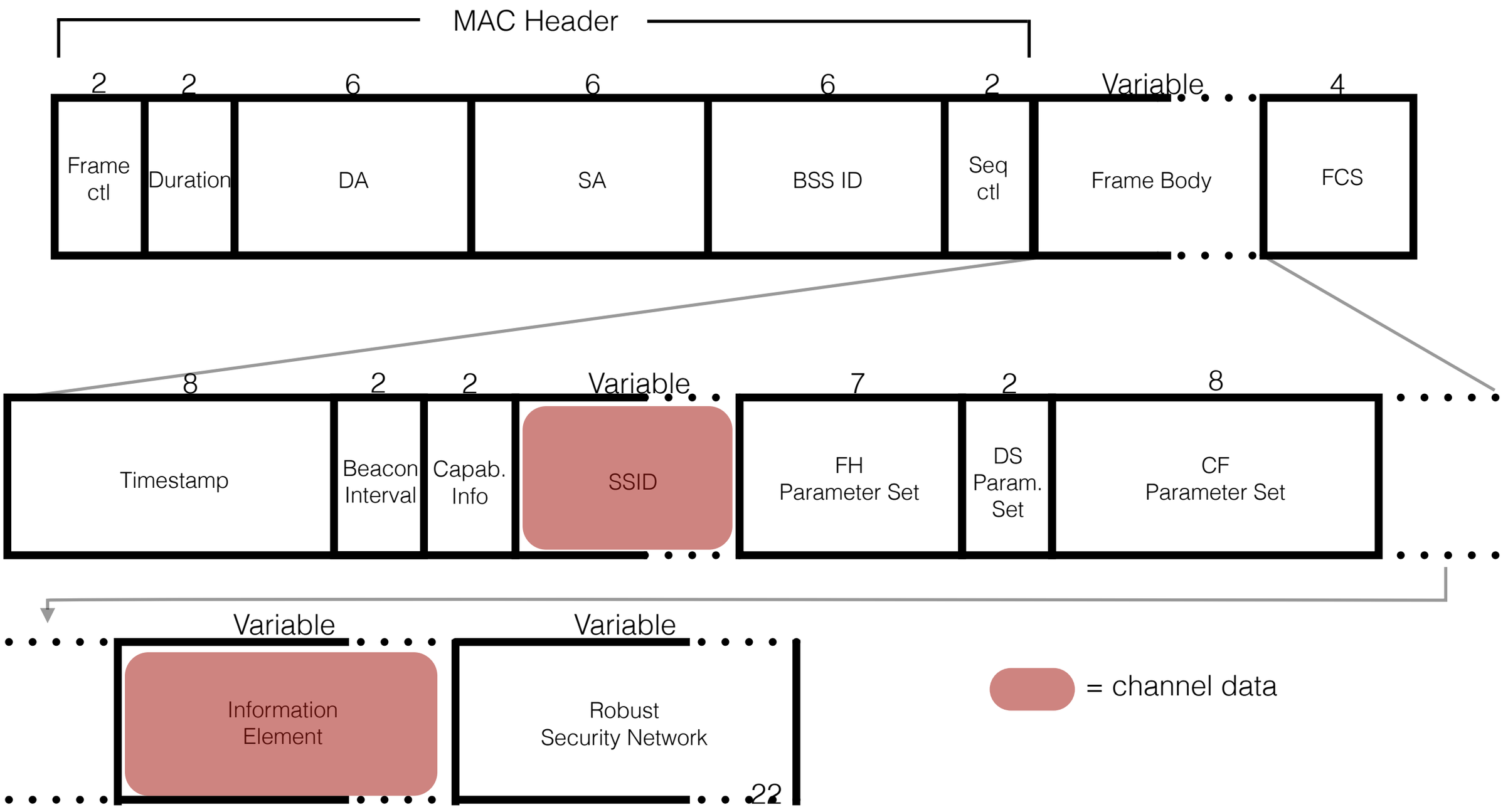
Covert Channel Design

Probe Request Frames



Covert Channel Design

Probe Response Frames



Covert Channel Design
Considerations

- Sometimes information elements cannot be injected.
  - Depends on the driver.
    - If available, channel bandwidth increases.
- Covert channel packet size is limited
  - 32 Bytes if only SSID Information Element is controlled.
  - ~255 Bytes if arbitrary IE is controlled.

Reading Data on Win XP

```
DWORD WINAPI WlanGetAvailableNetworkList(

    __in    HANDLE hClientHandle,

    __in    const GUID *pInterfaceGuid,

    __in    DWORD dwFlags,

    __reserved    PVOID pReserved,

    __out    PWLAN_AVAILABLE_NETWORK_LIST *ppAvailableNetworkList);
```

Reading Data on Win XP

```
typedef struct _WLAN_AVAILABLE_NETWORK_LIST {
        DWORD    dwNumberOfItems;

        DWORD    dwIndex;

        WLAN_AVAILABLE_NETWORK    Network[1];

} WLAN_AVAILABLE_NETWORK_LIST, *PWLAN_AVAILABLE_NETWORK_LIST;
```

Reading Data on Win XP

typedef struct _WLAN_AVAILABLE_NETWORK {

…

DOT11_SSID dot11Ssid;

…

} WLAN_AVAILABLE_NETWORK, *PWLAN_AVAILABLE_NETWORK;

Reading Data after Win XP

```
DWORD WINAPI WlanGetNetworkBssList(
        __in    HANDLE hClientHandle,
        __in    const GUID *pInterfaceGuid,
        __opt    const  PDOT11_SSID pDot11Ssid,
        __in    DOT11_BSS_TYPE
        __in    BOOL bSecurityEnabled,
        __reserved PVOID    pReserved,
        __out    PWLAN_BSS_LIST *ppWlanBssList);
```

Reading Data after Win XP

```
typedef struct _WLAN_BSS_LIST {
        DWORD    dwTotalSize;

        DWORD    dwNumberOfItems;

        WLAN_BSS_ENTRY    wlanBssEntries[1];

} WLAN_BSS_LIST, *PWLAN_BSS_LIST;
```

Reading Data after Win XP

typedef struct _WLAN_BSS_ENTRY {

DOT11_SSID dot11Ssid;

…

DOT11_MAC_ADDRESS dot11Bssid;

…

ULONG ulleOffset;

ULONG ulleSize;

} WLAN_BSS_ENTRY, *PWLAN_BSS_ENTRY;

Demo

Reading data “from the air”

Injecting Data

```
DWORD WINAPI WlanScan(

    __in HANDLE hClientHandle,

    __in const GUID *pInterfaGuid,

    __in_opt const PDOT11_SSID pDot11Ssid,

    __in_opt const PWLAN_RAW_DATA pIeData,

    __reserved PVOID pReserved);
```

Demo

Writing data “to the air”

Summary

PoC covert channel between a compromised host and an attacker

- Win Vista - 7 through Native API
- Can coexist with active WiFi connections
- Difficult to discover, unless actively (manually) looking for it
- Can serve as fallback from other “connect from” payloads
- Bypass network “boundaries”

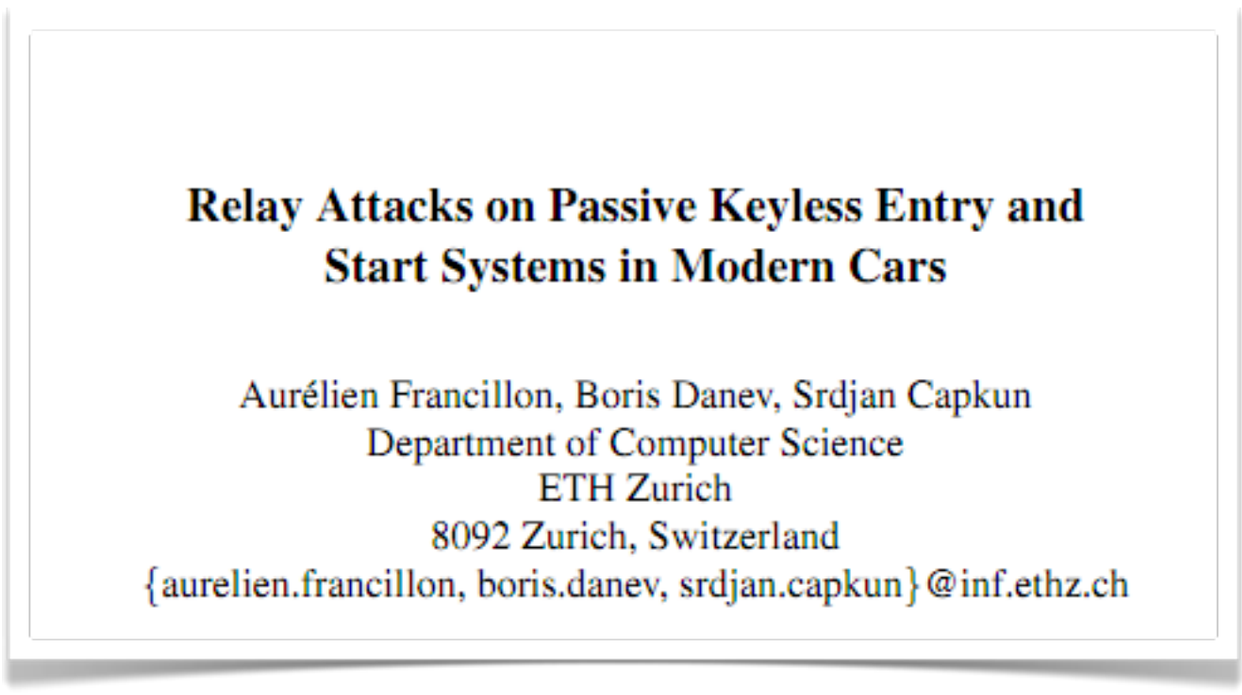
Conclusions

- WiFi covert channels are useful as post-exploitation fallback methods.
- Active client-side attacks can also deploy a wireless covert channel endpoint.
- The Windows Native WiFi API, by design, allows covert communications with low privileges.

Conclusions

- Perimeter is **gone**, wireless vectors such as bluetooth and WiFi will evolve with “device” evolution.

http://eprint.iacr.org/2010/332.pdf



Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars

Aurélien Francillon, Boris Danev, Srdjan Capkun
Department of Computer Science
ETH Zurich
8092 Zurich, Switzerland
{aurelien.francillon, boris.danev, srdjan.capkun}@inf.ethz.ch

Future work & enhancements

- Evolve prototype to a usable full covert channel
- Work out WinXP availability
- Many-to-one communication (many clients to one attacker) - Multiplexing
- Encryption

Questions

Mini-challenge

A Windows host will be broadcasting a secret message.
Find the secret message and win a **Mate combo**

Contact:
(ablanco|egutesman) [a7] coresecurity [d07] com