CS 114: Network Security

Lecture 15 - Wireless

Prof. Daniel Votipka

Spring 2023

(some slides courtesy of Prof. Micah Sherr)

Tufts University

Plan for today

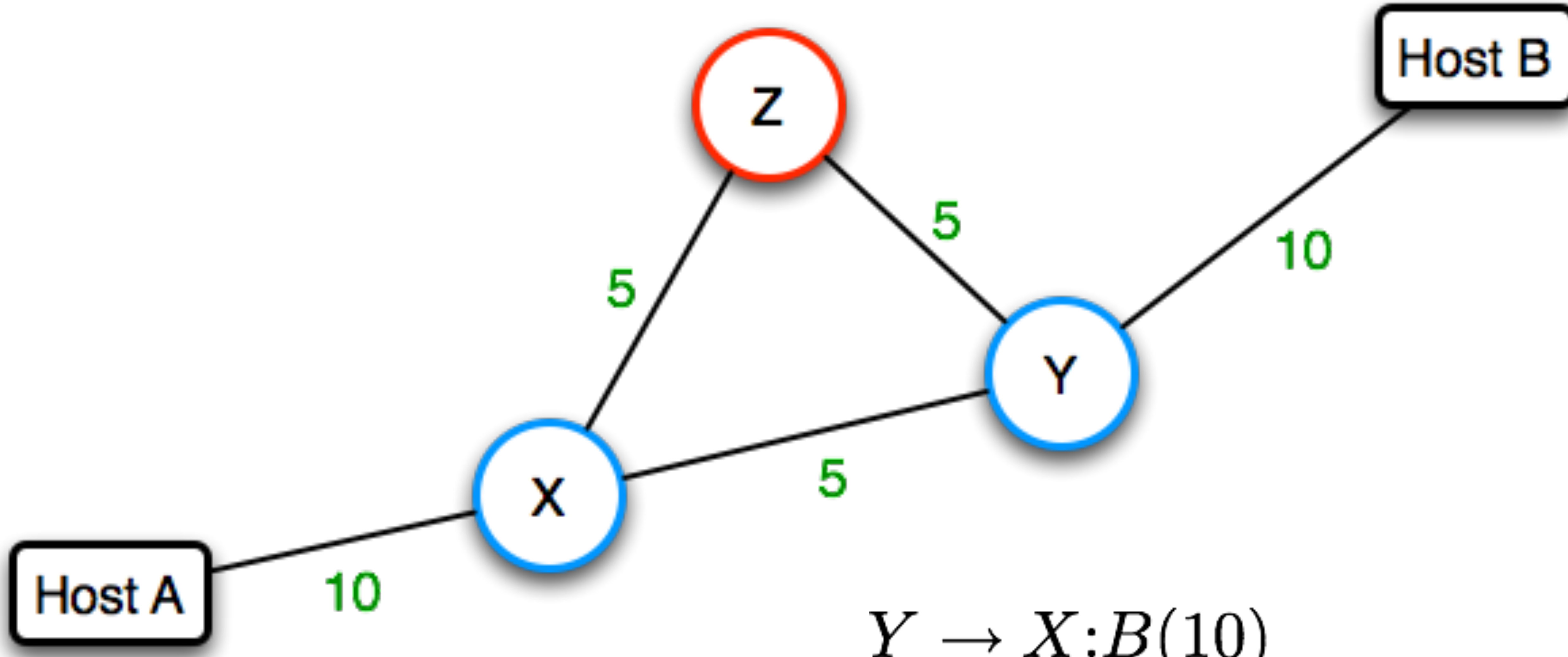
- Review Routing
  - Filtering with RPKI
- Secure Wireless
  - Overview
  - Protocol - 802.11
  - Attacks/Defenses

Administrivia

- HW1p3 Hint:
  - nc -l 9999 (listen on port 9999 and print to stdout)
- My Thursday office hours are cancelled this week

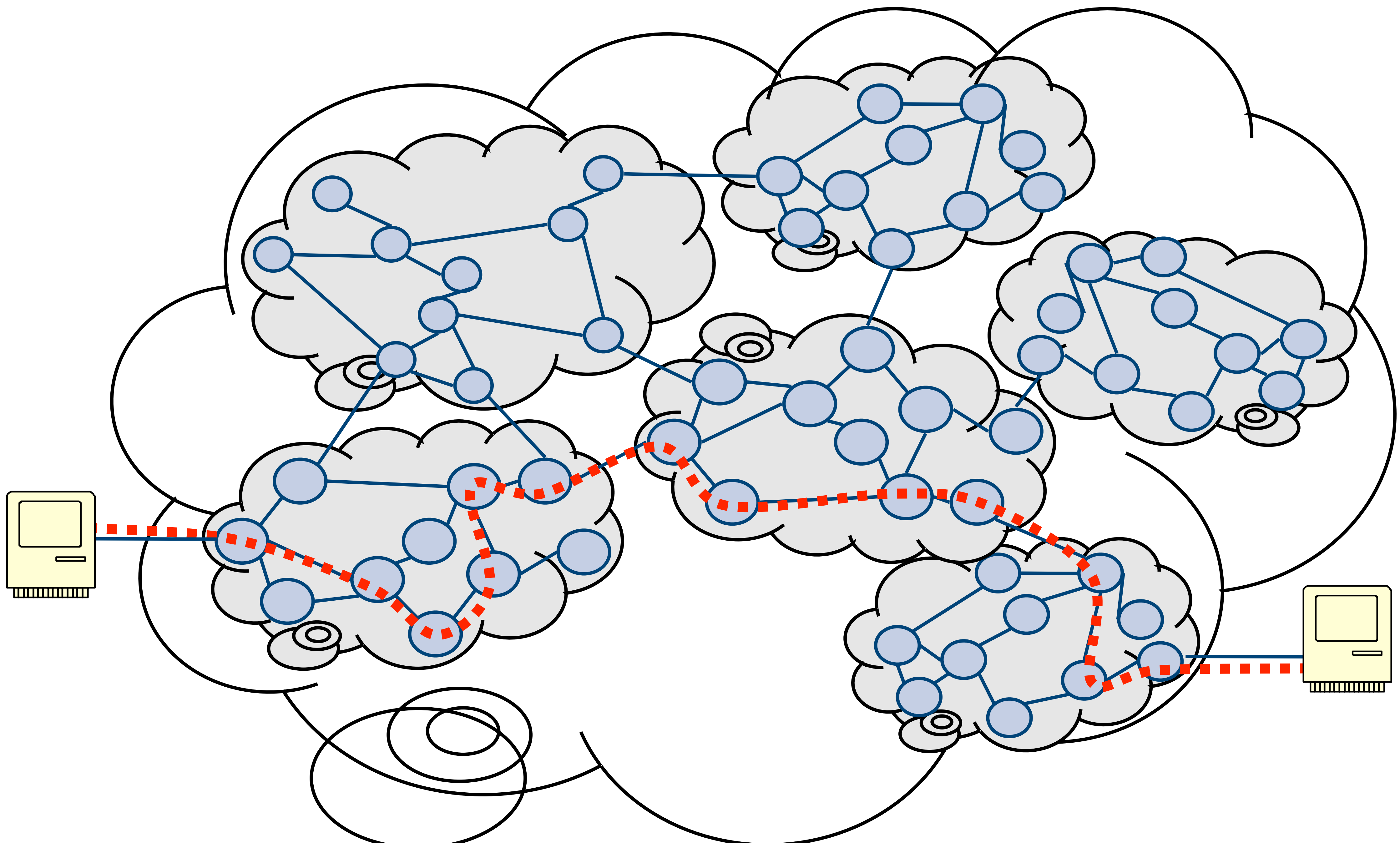
Routing Review

Normal Behavior



$$Y \rightarrow X{:}B(10)$$
$$Y \rightarrow Z{:}B(10)$$
$$Z \rightarrow X{:}Y(5), B(15)$$
$$X \rightarrow A{:}Z(5), Y(5), B(15)$$

- Each AS is responsible for moving packets inside it.
- Intra-AS routing is (mostly) independent from Inter-AS routing.



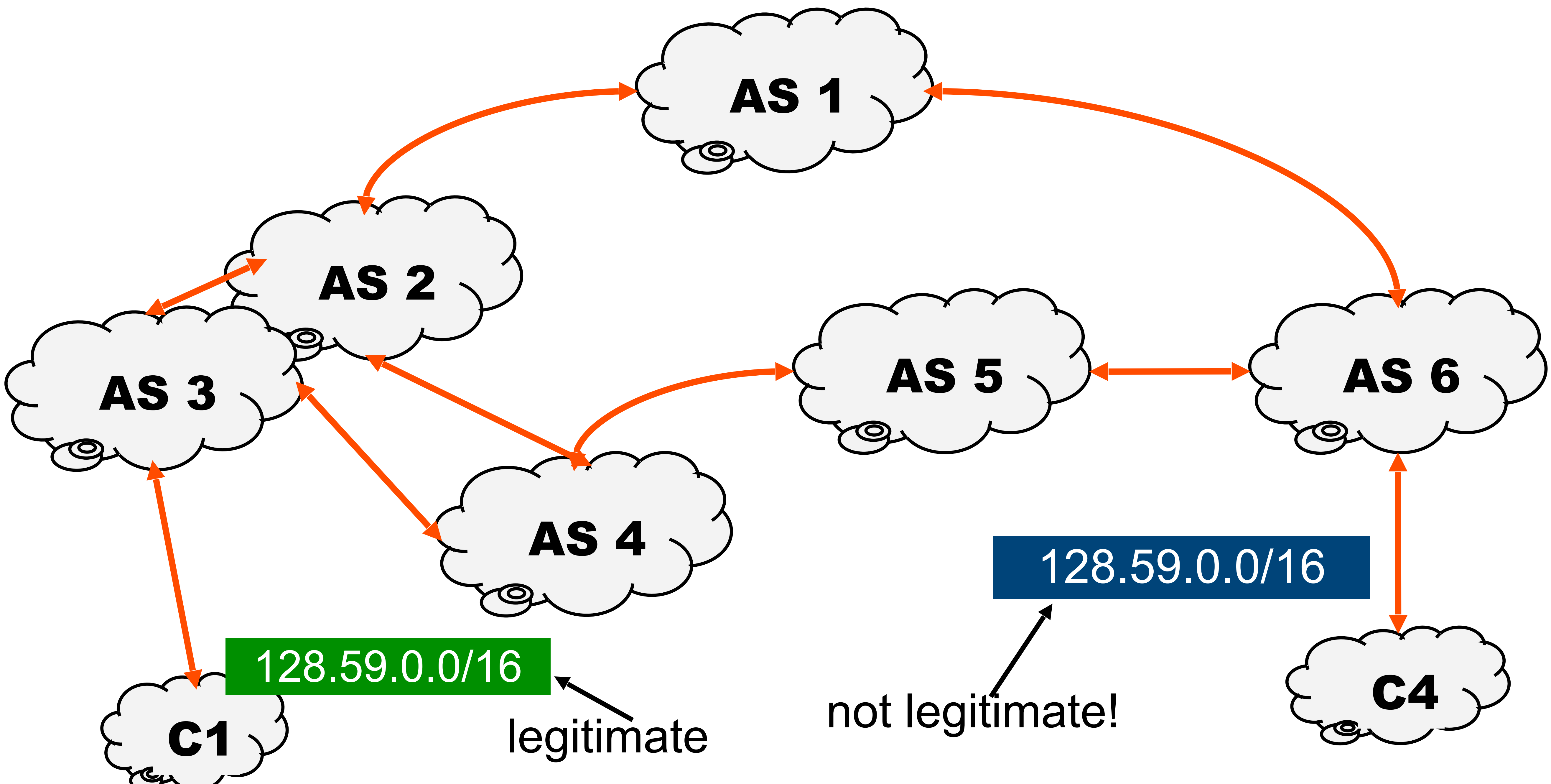
The BGP Protocol

- BGP messages
  - **Origin** announcements:
    - "I own this block of addresses"
  - Route **advertisements**:
    - "To get to this address block, send packets destined for it to me. And by the way, here is the path of ASes it will take"
  - Route **withdrawals**:
    - "Remember the route to this address block I told you about, that path of ASes no longer works"

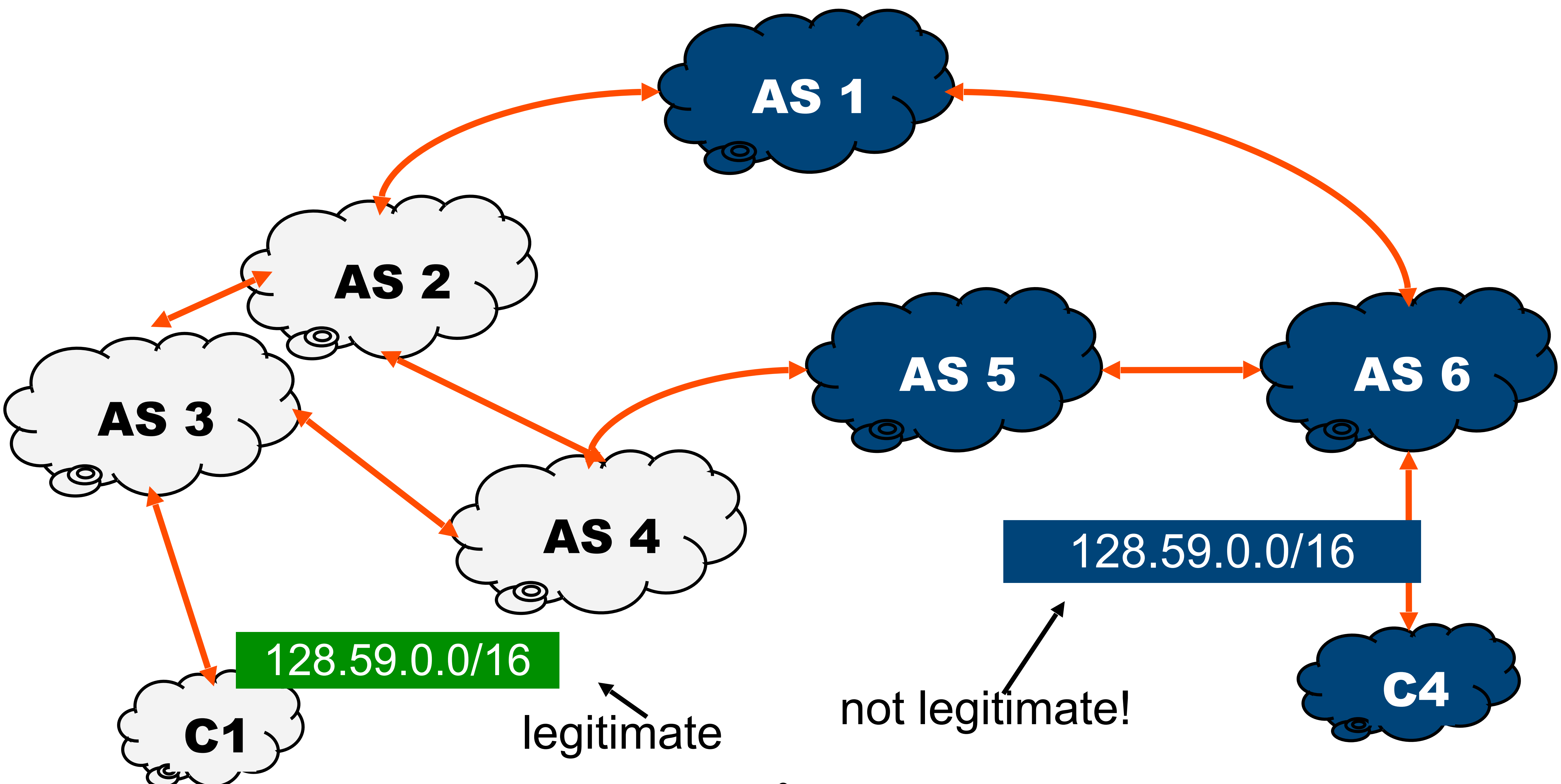
- Route decisions
  - Border routers receive origin announcements/route advertisements from their peers
  - They choose the "best" path and send their selection downstream
- BGP Attributes
  - BGP messages have additional attributes to help routers choose the "best" path

CIDR Block	Path			Attributes
123.125.28.0/24	768	4014	664	bkup

- If another AS advertises one of our prefixes, bad things happen:



- Prefix becomes unreachable from the part of the net believing C4's announcement.



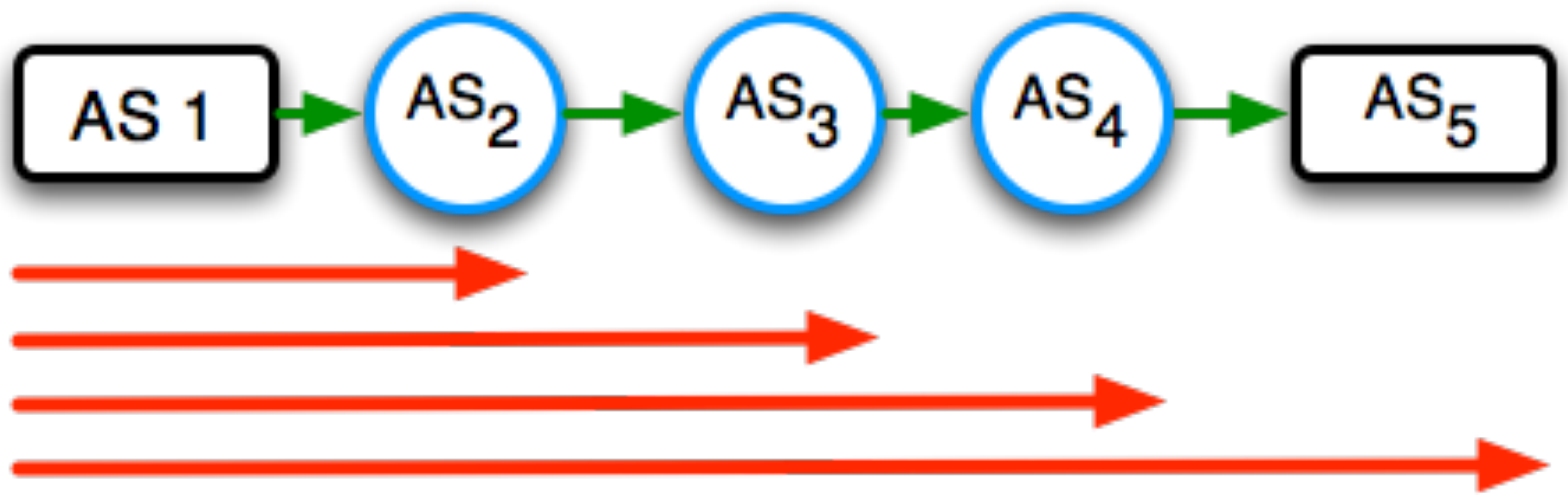
Attack: Longest-Prefix Matching

- Within the AS, a prefix can be broken into smaller blocks and advertised as such
- Because of **longest-prefix matching**, these will be preferred (eg. 12.10.8.0/24 is preferred over 12.0.0.0/8 because it is more specific)
- Attacker can get clever (say 100.200.0.0/16 is targeted IP block)
  - Attacker sends origin announcement for 100.200.0.0/17 and 10.200.129.0/17 (covers all of 100.200.0.0/16!)
  - Attack has limits: most ASes won't propogate announcements more specific than /24

Filtering

- Filtering just drops BGP message (typically advertisements) as they are passed between ASes
  - Ingress filtering (as it is received)
  - Egress filtering (as it is sent)
- Types of filtering
  - By prefix (e.g., *bogon*/*martian* list)
  - By path
  - By policy
- ISP ASes aggressively filter (this is the main security mechanism)

Route Attestations

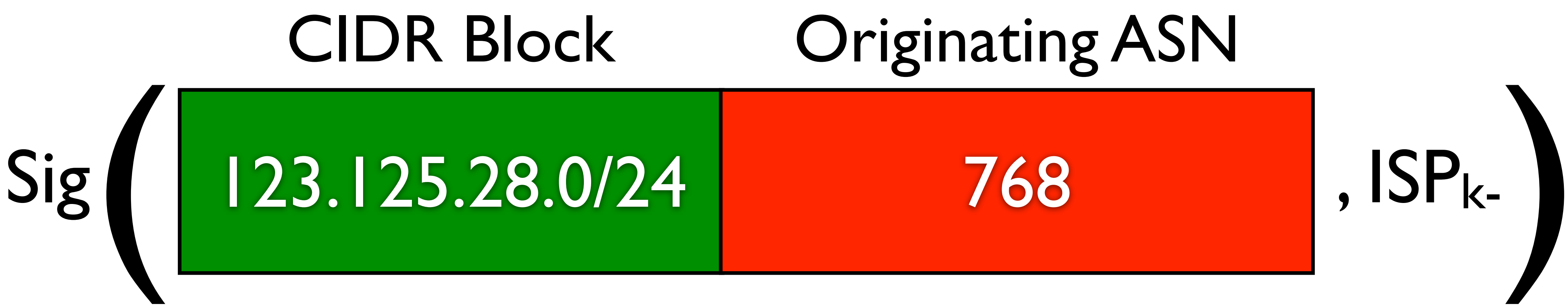


- Signing recursively: each advertisement signs everything it receives, plus the last hop.

$$(5, (4, (3, (2, 1)_{k_{AS_1}})_{k_{AS_2}})_{k_{AS_3}})_{k_{AS_4}}$$

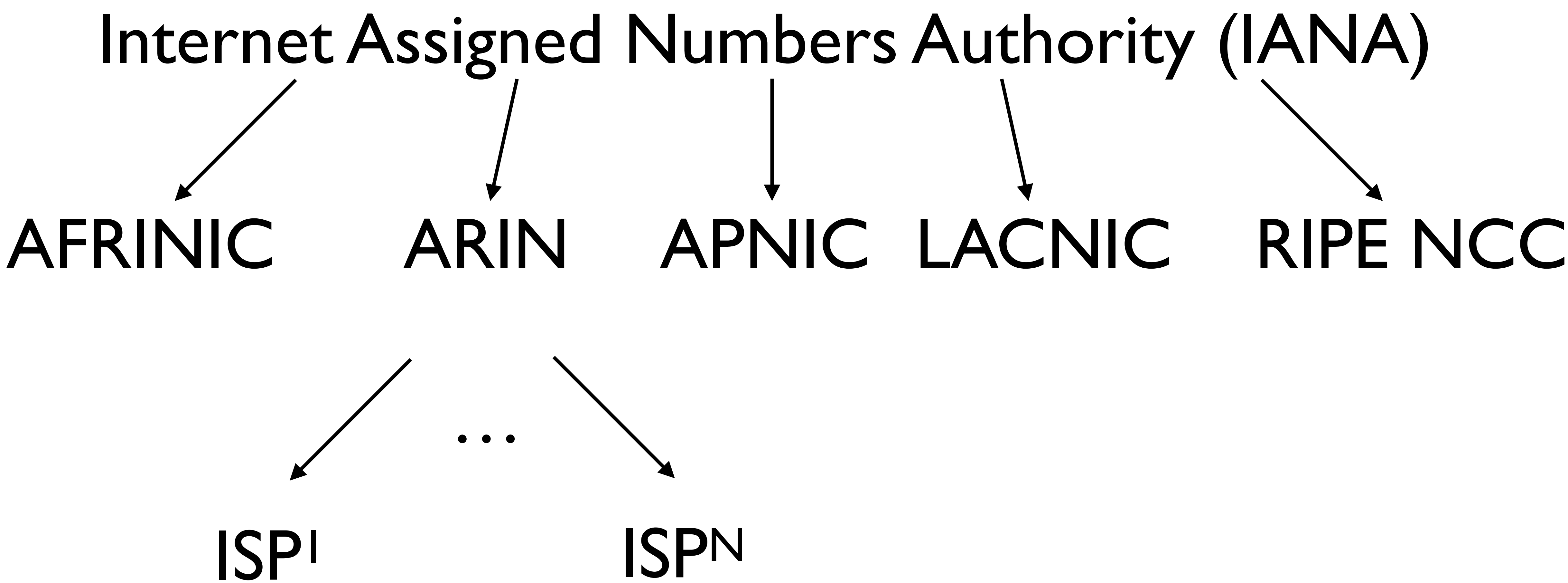
Filtering with RPKI

https://www.rfc-editor.org/rfc/rfc6480

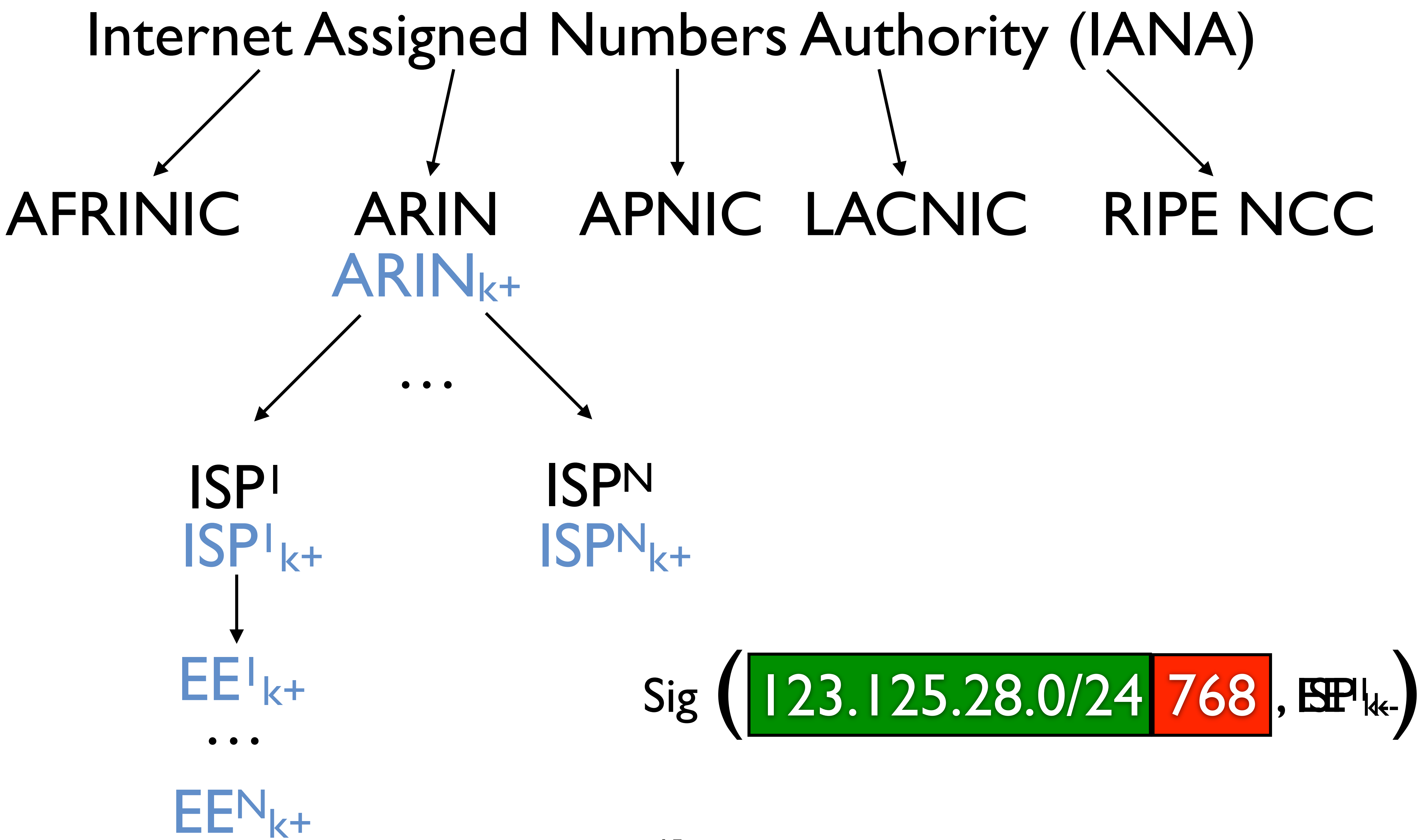


- ISPs publish signed route originations
- Other ISPs use signed routes to filter BGP route advertisements

IP address allocation



Resource PKI



RPKI Repository

ISP[1]

Sig (123.125.28.0/24 | 768 , EE^1_{k-})

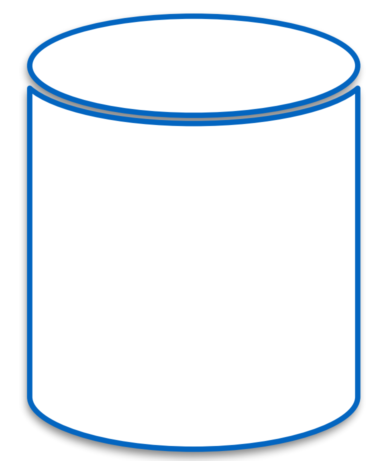
Sig (123.125.29.0/24 | 768 , EE^2_{k-})

Sig (123.125.40.0/24 | 768 , EE^5_{k-})

Sig (revoke EE^3_{k+} and EE^4_{k+} , EE^6_{k-})

Sig (ROA_1, SHA(ROA_1)
ROA_2, SHA(ROA_2)
ROA_5, SHA(ROA_5)
CRL, SHA(CRL) , EE^7_{k-})

ARIN

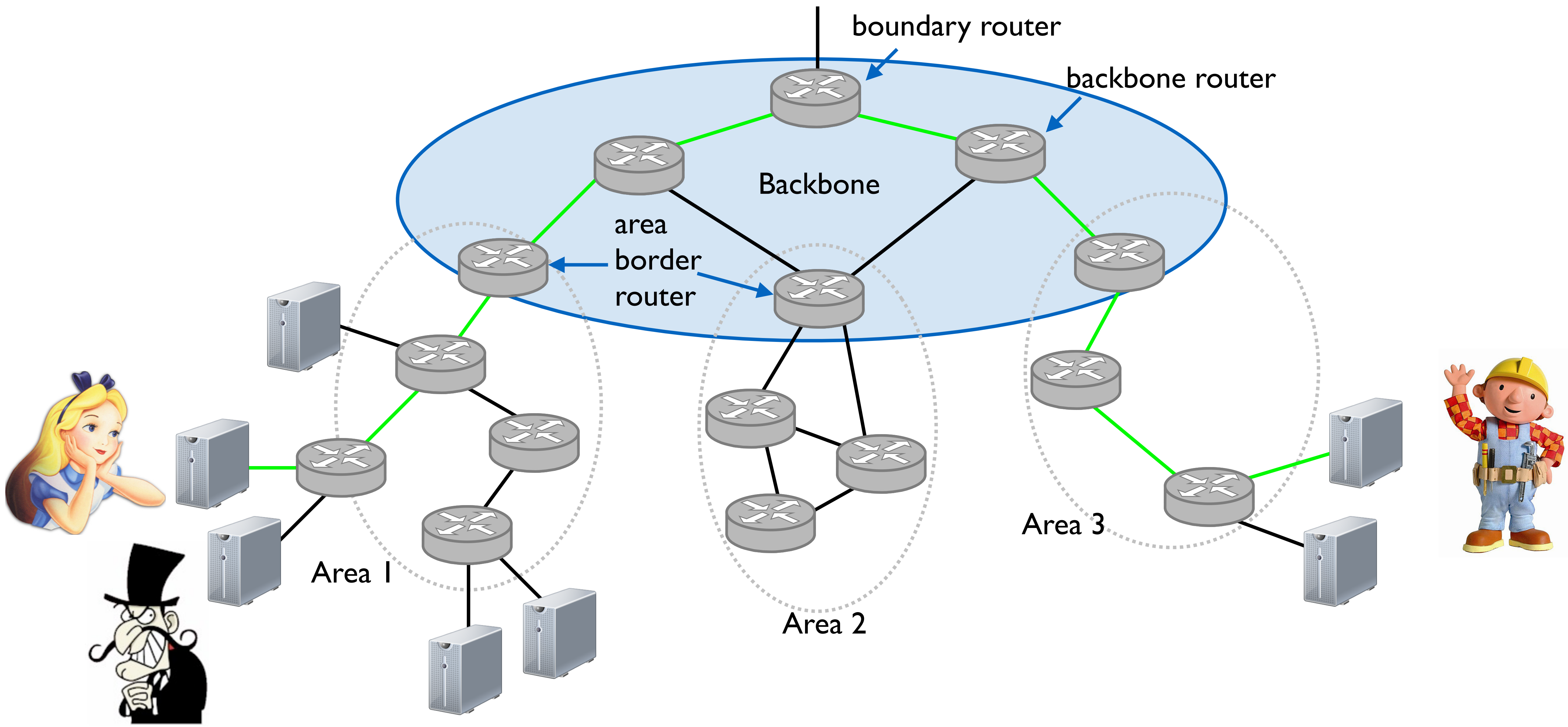


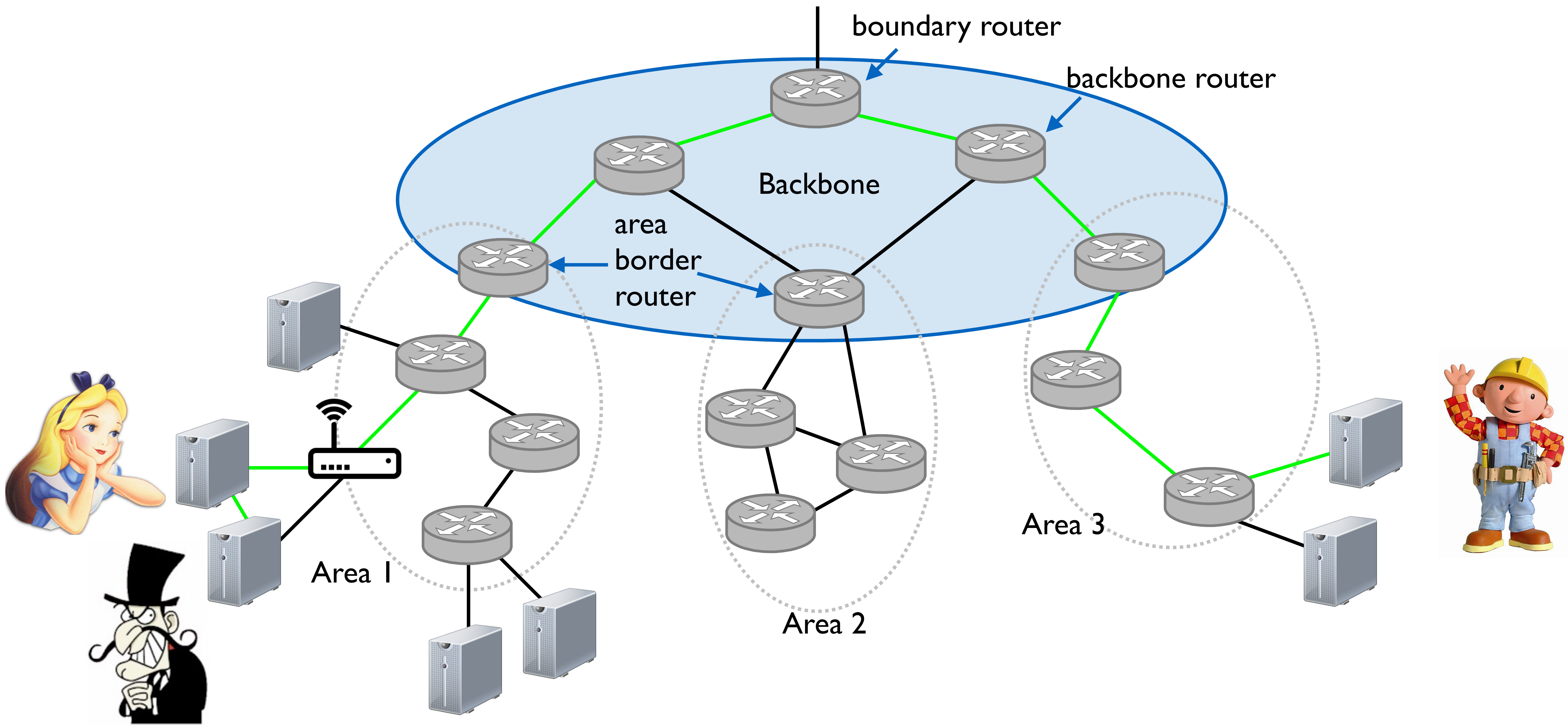
Plan for today

- Review Routing
  - Filtering with RPKI
- Secure Wireless
  - Overview
  - Protocol - 802.11
  - Attacks/Defenses

Wireless makes network security much more difficult

- Wired:
  - If Alice and Bob are connected via a wire, Eve can only eavesdrop if she has physical access to that wire* (exceptions?)
- Wireless:
  - Everybody shout (broadcast) as loud as you can
  - Everyone is eavesdropping


boundary router
backbone router
Backbone
area
border
router
Area 1
Area 2
Area 3


boundary router
backbone router
Backbone
area
border
router
Area 1
Area 2
Area 3

Plan for today

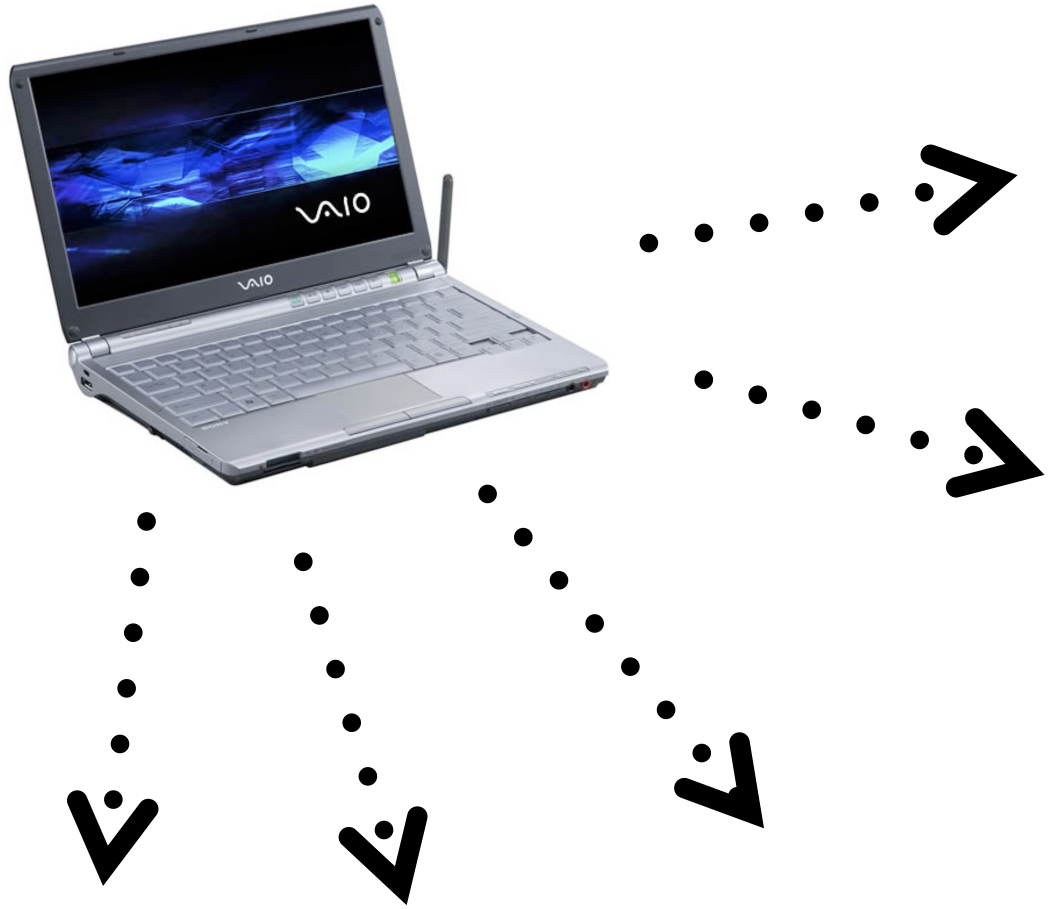
- Review Routing
  - Filtering with RPKI
- Secure Wireless
  - Overview
  - Protocol - 802.11
  - Attacks/Defenses

Wireless Networking: 50,000 ft view

- Protocols defined in IEEE 802.11 standards
- Access points (APs) may periodically broadcast *beacon frames* to advertise its presence (and some configuration parameters)
- Authentication:
  - client sends *authentication frame* to AP
  - if successful, client sends *association request frame* to AP, requesting allocation of resources
    - if successful, AP responds with *association response frame*
- Data sent via *data frames*
- Session Termination:
  - AP sends *disassociation frame* and *deauthentication frame*

Plan for today

- Review Routing
  - Filtering with RPKI
- Secure Wireless
  - Overview
  - Protocol - 802.11
  - Attacks/Defenses



Unsecured wireless:

Problem #1:

Everybody is the receiver.

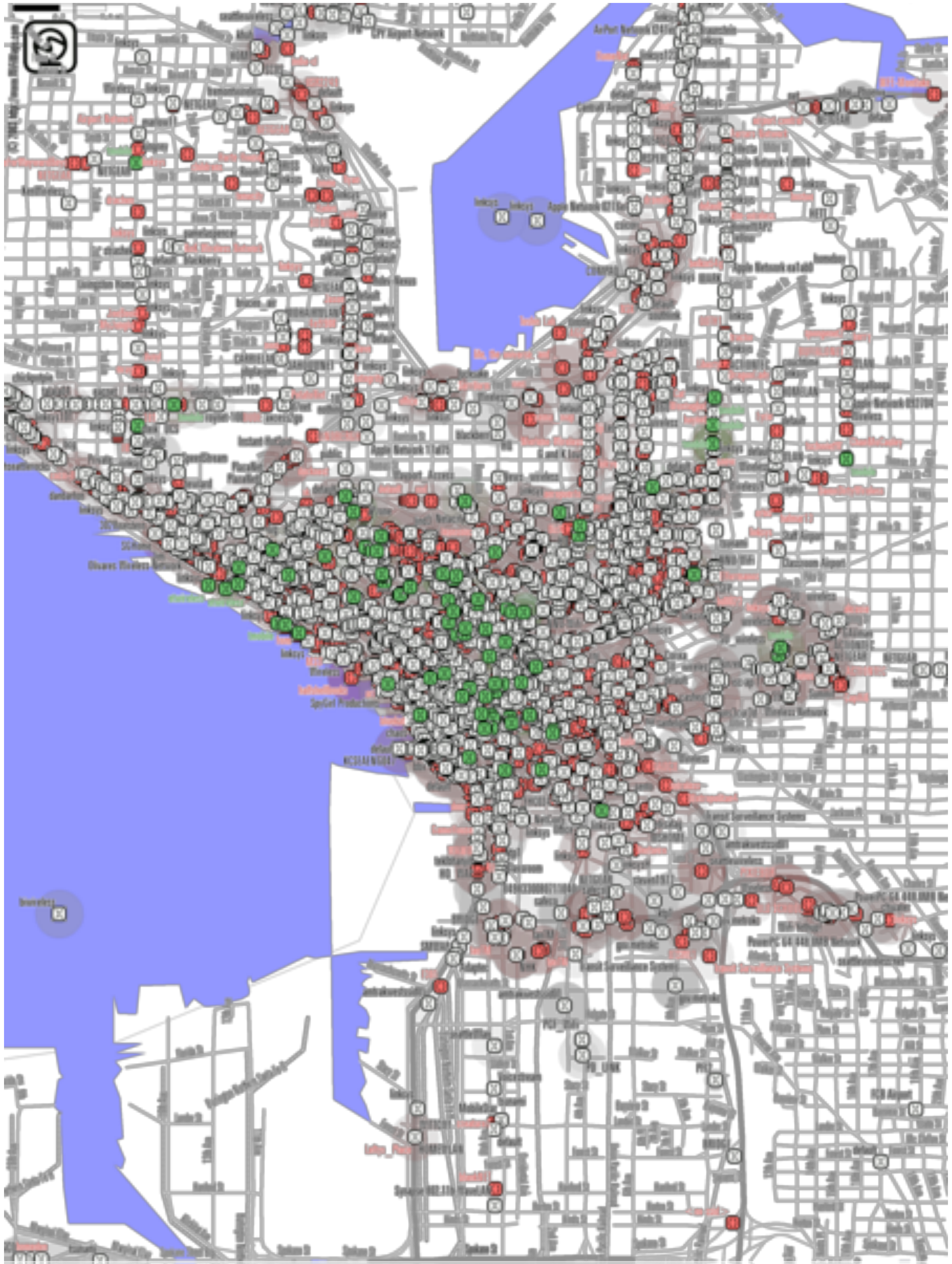
Unsecured wireless:

Problem #2:

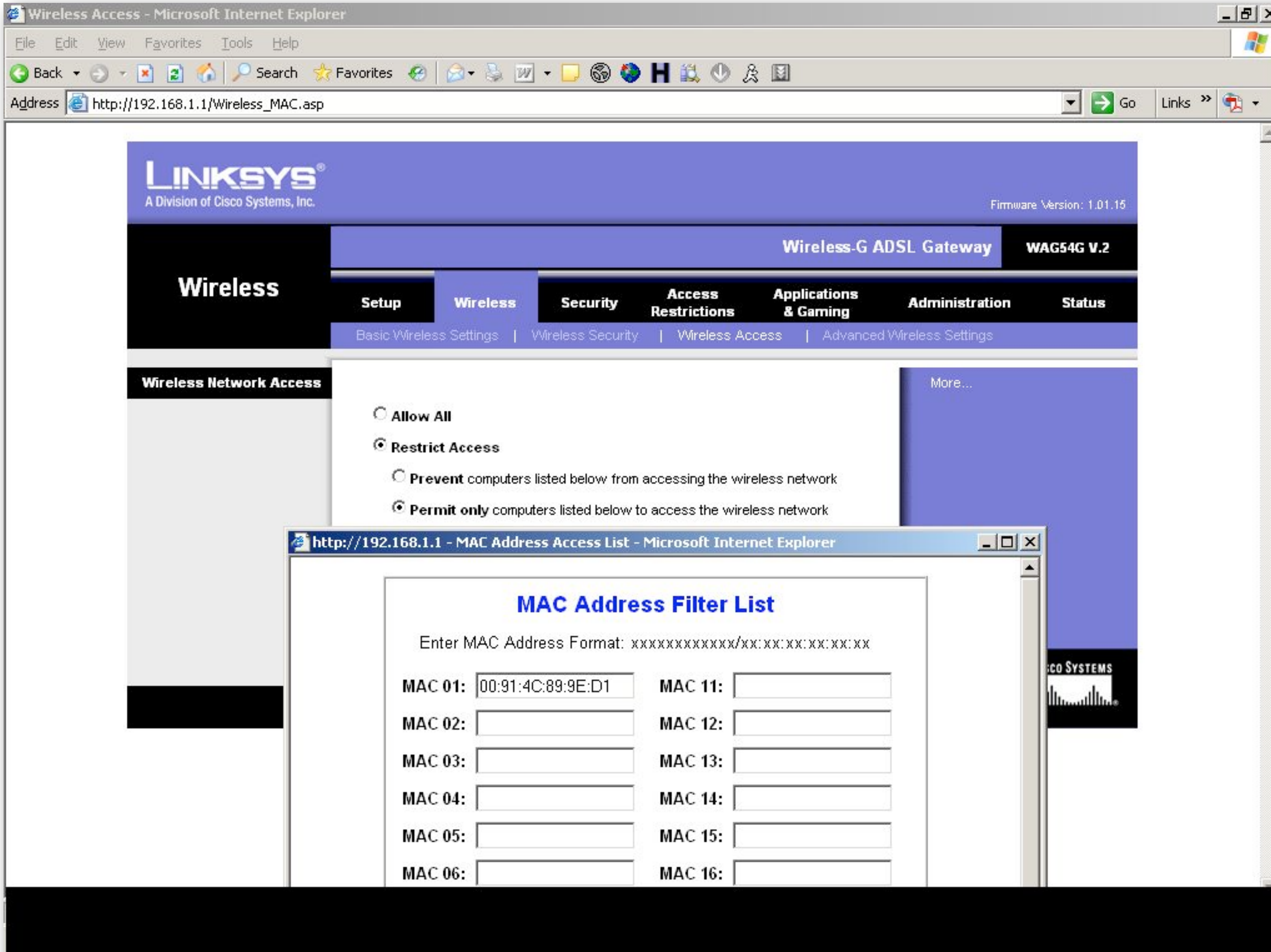
Any one can join.

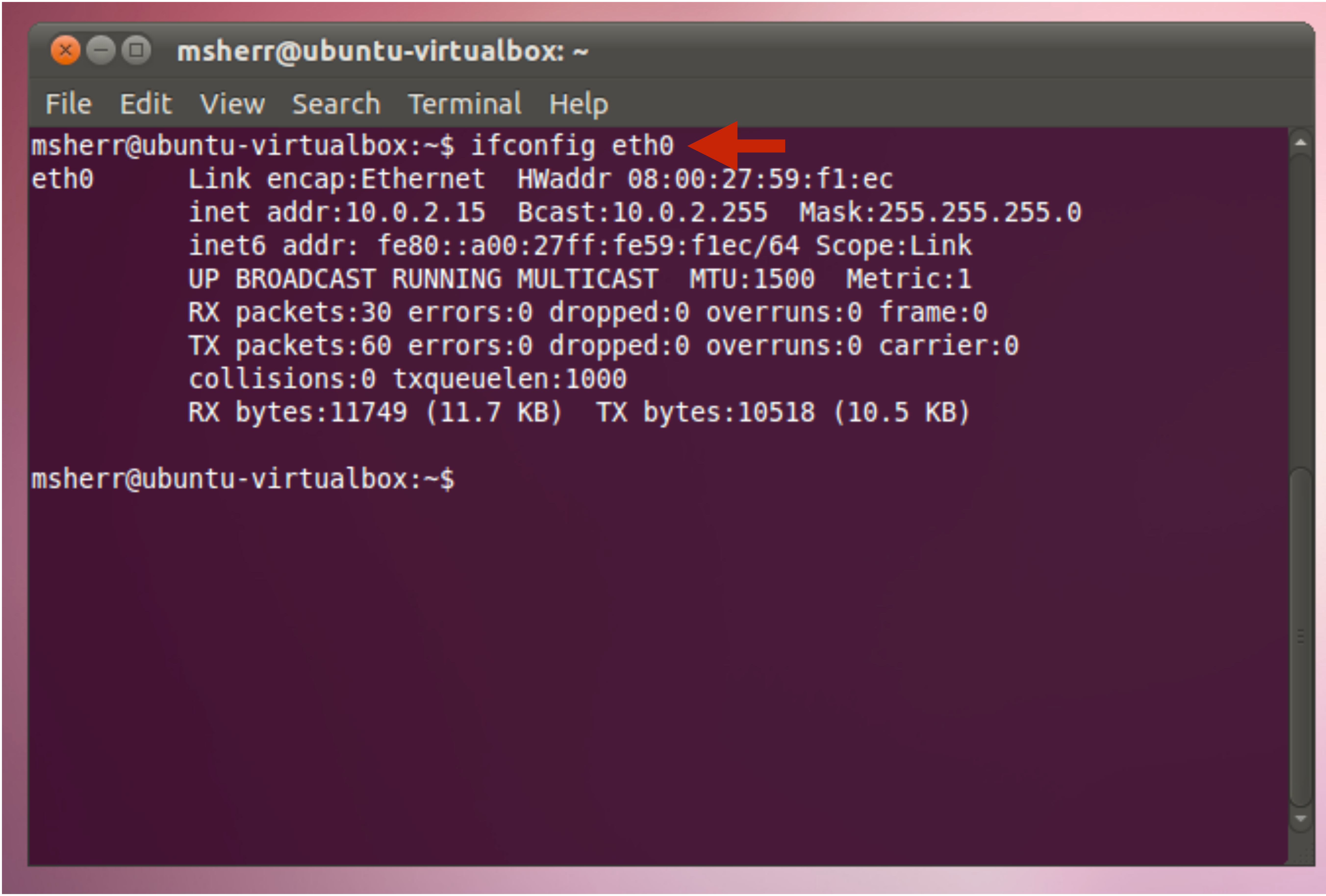
Finding wireless networks is easy

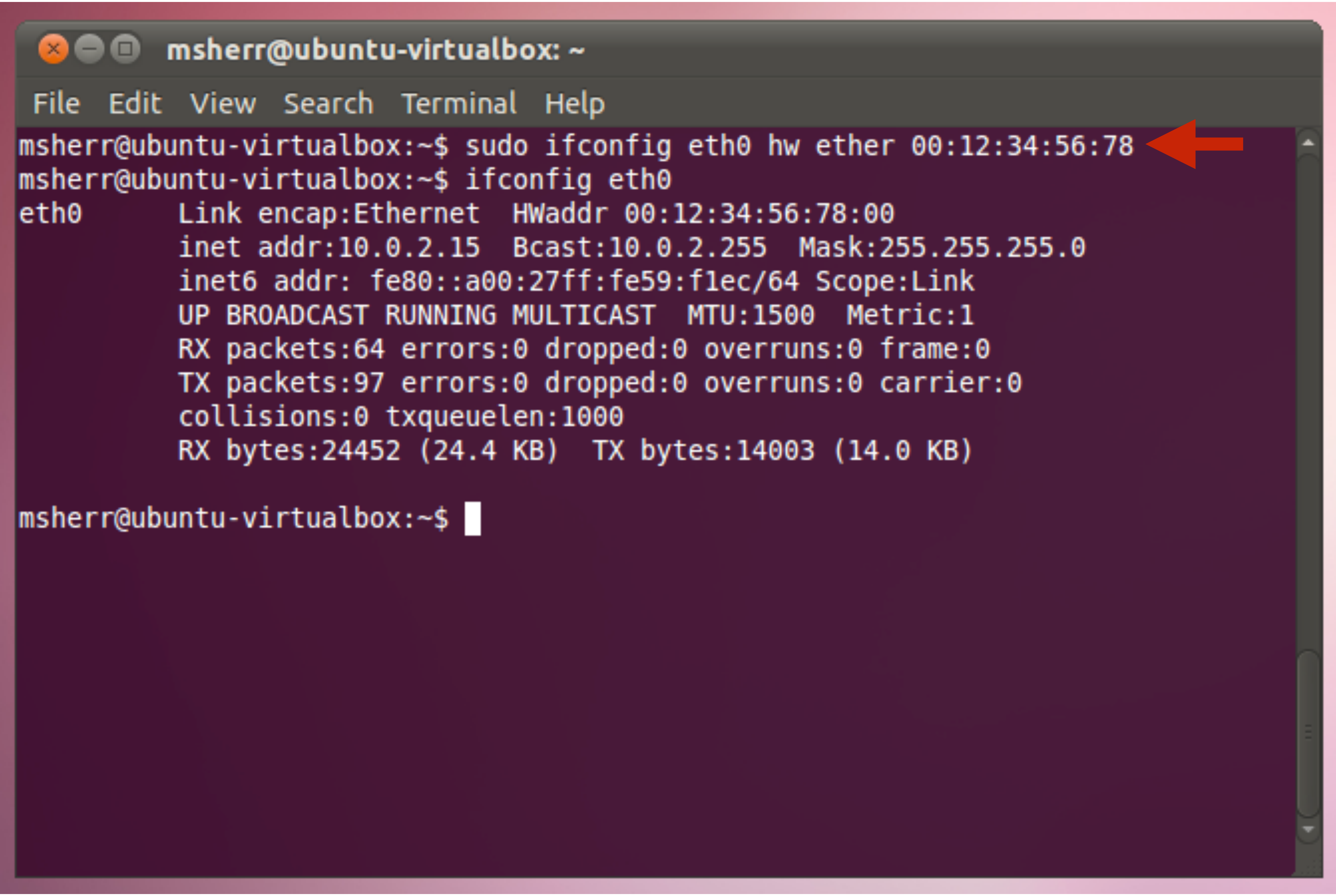
- wardriving
- warbiking
- warwalking
- warrailing



MAC Filtering




msherr@ubuntu-virtualbox: ~
File Edit View Search Terminal Help
msherr@ubuntu-virtualbox:~$ ifconfig eth0
eth0 Link encap:Ethernet HWaddr 08:00:27:59:f1:ec
inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fe59:f1ec/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:30 errors:0 dropped:0 overruns:0 frame:0
TX packets:60 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:11749 (11.7 KB) TX bytes:10518 (10.5 KB)
msherr@ubuntu-virtualbox:~$


msherr@ubuntu-virtualbox: ~
File Edit View Search Terminal Help
msherr@ubuntu-virtualbox:~$ sudo ifconfig eth0 hw ether 00:12:34:56:78
msherr@ubuntu-virtualbox:~$ ifconfig eth0
eth0 Link encap:Ethernet HWaddr 00:12:34:56:78:00
inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::a00:27ff:fe59:f1ec/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:64 errors:0 dropped:0 overruns:0 frame:0
TX packets:97 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:24452 (24.4 KB) TX bytes:14003 (14.0 KB)
msherr@ubuntu-virtualbox:~$

SSID hiding

- APs broadcast **Service Set Identifiers (SSIDs)** to announce their presence
- In theory, these should identify a particular wireless LAN
- In practice, SSID can be anything that's 2-32 octets long
- To join network, client must present SSID
- Security mechanism for preventing interlopers:
  - Don't advertise SSID
  - Problem:
    - To join network, client must present SSID
    - This is not encrypted, even if network supports WEP or WPA

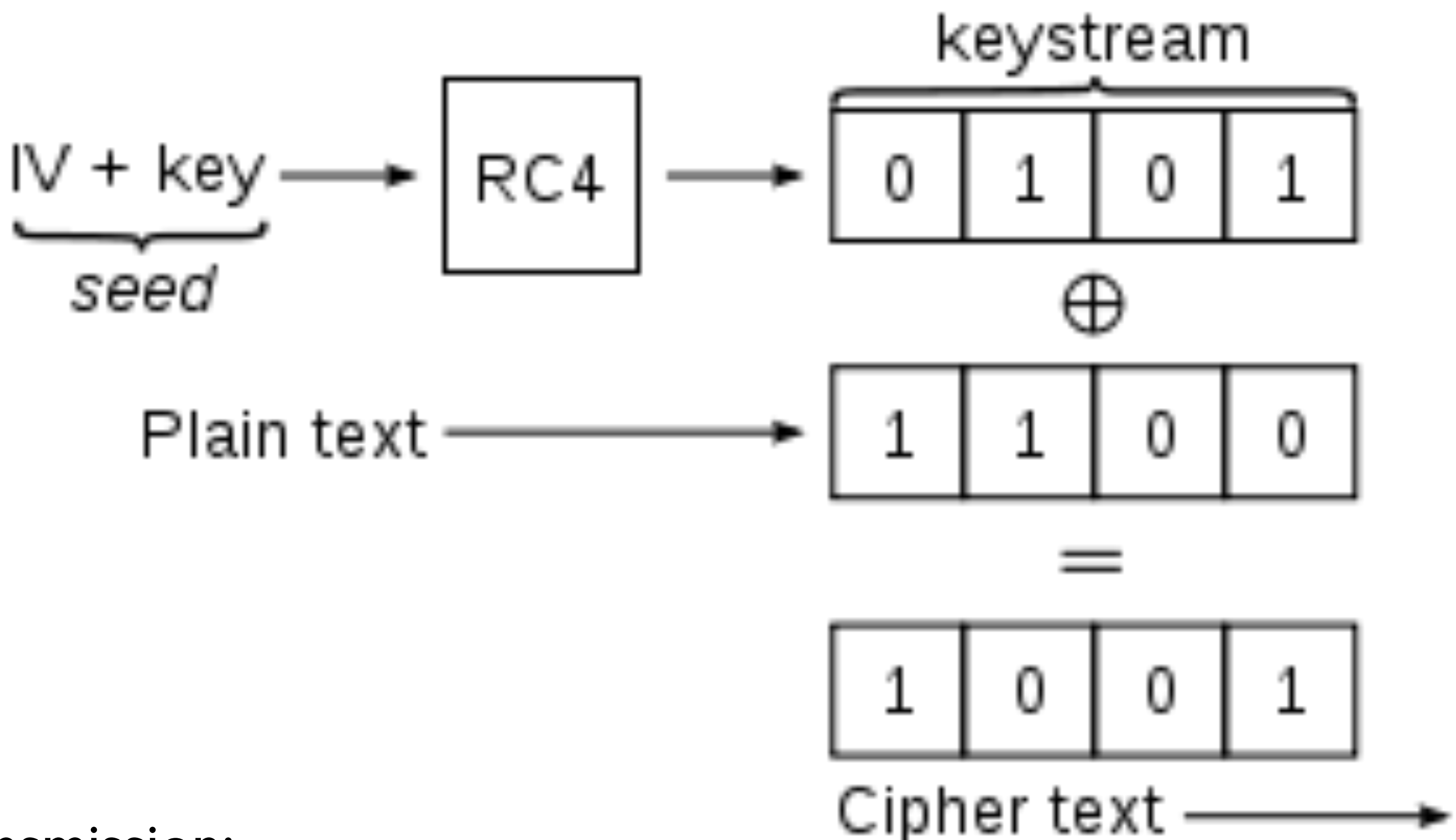
Wireless Security

Let's sprinkle on some of that crypto magic sauce

Wired Equivalent Privacy (WEP)

- Part of original 802.11 standard
- Uses stream cipher:
  - WEP uses RC4 - supports seed up to 256 bits
    - seed = 24-bit IV + WEP key
- In WEPv1, key was 40 bits ➡ 64bit seed
- Later versions supported seeds of 128 and 256 bits

Wired Equivalent Privacy (WEP)



- Data transmission:
  - Produce keystream S using RC4 with seed function f(K,IV)
  - $C = M \oplus S$
  - send (IV, C) frames
  - knowledge of IV and K sufficient to decrypt C

WEP Authentication Modes

- **Open System:**
  - client doesn't need to provide any credentials
  - immediate association with access point
  - but can only send and receive info if using correct key
- **Shared Key:**
  - client must prove knowledge of WEP key before associating
  - AP sends client plaintext challenge; response is challenge encrypted with the correct key
- Q: Which is more secure?

WEP Shared Key Vulnerability

- Random Challenge: “`jk4533hfdsa9`”
- Response: {IV, “`jk4533hfdsa9`” ⊕ RC4(K,IV)}
  - here, RC4(K,IV) denotes RC4 encryption using a key derived from key K and IV
- Eavesdropper can observe plaintext challenge and encrypted response, and can produce:
  - challenge ⊕ response = RC4(K,IV)
  - RC4(K,IV) sufficient to authenticate:
    - next challenge: “`abcdef`”
    - Eve responds (without knowing K!): {IV, “`abcdef`” ⊕ RC4(K,IV)}

WEP Problems: IV Collisions

- IVs are too small... likely collision(s) after a few hours
  - when IVs are the same, two ciphertexts can be xor'ed together to produce the xor of the plaintexts
    - statistical analysis will then yield plaintexts
      - redundancy in IP packets makes this easy!
      - knowledge of protocols further limits the possibilities
      - or, attacker sends message thru Internet to a wireless client in a manner that will result a known response (e.g., ping message)
  - if multiple messages share same IV, once one is recovered, others can be trivially/immediately recovered --**WHY?**

WEP Problems: Exploiting RC4 Weaknesses

- RC4 has a weakness: first few bytes of keystream are sometimes not particularly random looking [Fluhrer, Mantin and Shamir Attack; 2001]
- Mathematical result: Given enough keystreams, it's possible to construct the key [ciphertext-only attack]
- Attacker's goal: Get a lot of keystreams!
  - Basic approach: replay a bunch of ARP packets
  - AP will respond to replayed ARP
  - Sufficient number of AP's encrypted packets will yield key
- An aside: standard RC4 fix: discard first *n* bytes of keystream (usually $n \geq 3072$)

Story Time: TJX Data Breach

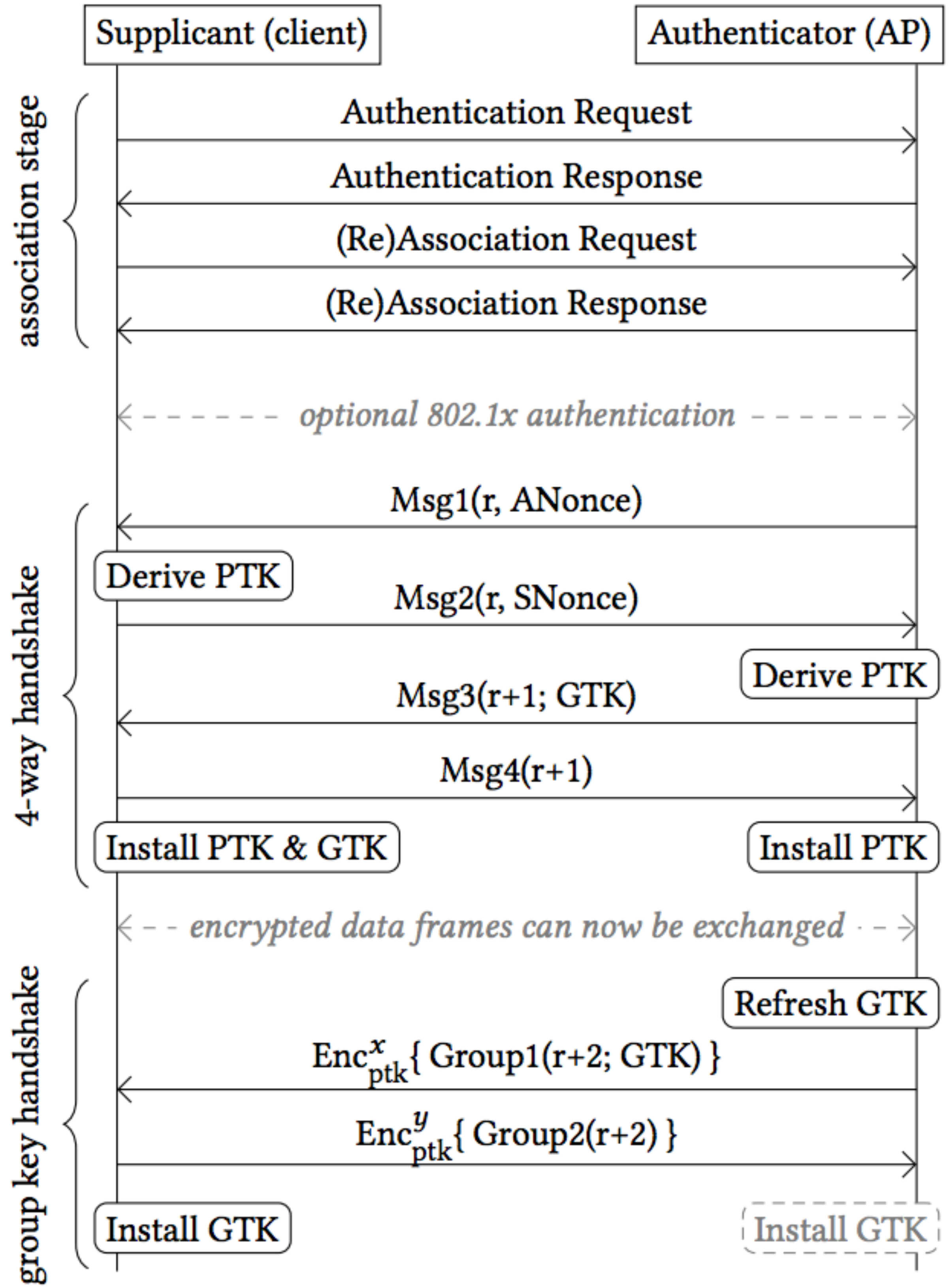
- TJX (TJMaxx + Marshalls + Bob's) main database compromised in 2007
  - ~94M credit and debit cards stolen
- Scanning devices, cash registers, and PCs in Minnesota Marshalls wirelessly communicated to server, which communicated to backend database
- Wireless data encrypted using WEP
- WEP key stolen from MN parking lot. Uh-oh.
- **Lesson: Don't use WEP!**

Wi-Fi Protected Access (WPA)

- Engineered to be the "secure replacement" for WEP
- Authentication stages:
  - Shared secret used to derive encryption keys
  - Client authenticates to AP
  - Encryption keys are used to produce keystreams for encrypting traffic

Pairwise Transit Key (PTK) = f(PSK, ANonce, SNonce, AP MAC address, STA MAC address)

Pre-Shared Key (PSK)



Wi-Fi Protected Access (WPA)

- Two Modes:
  - **PSK (Pre-shared Key):**
    - also called "WPA Personal"
    - shared secret manually entered into all devices
    - designed for home use
  - **802.1x Mode:**
    - also called "WPA Enterprise"
    - authentication handled by backend service (e.g., RADIUS server) via Extensible Authentication Protocol (EAP)
      - may make use of certificates or other authentication techniques
      - e.g., SaxaNet

Wi-Fi Protected Access (WPA)

- Encrypting Traffic (2 confidentiality protocols):
  - **Temporal Key Integrity Protocol (TKIP)**:
    - uses RC4, but designed to improve upon WEP's shortcomings
    - increases size of IV to 48 bits
    - rather than just concatenate IV, uses more complex key mixing routine

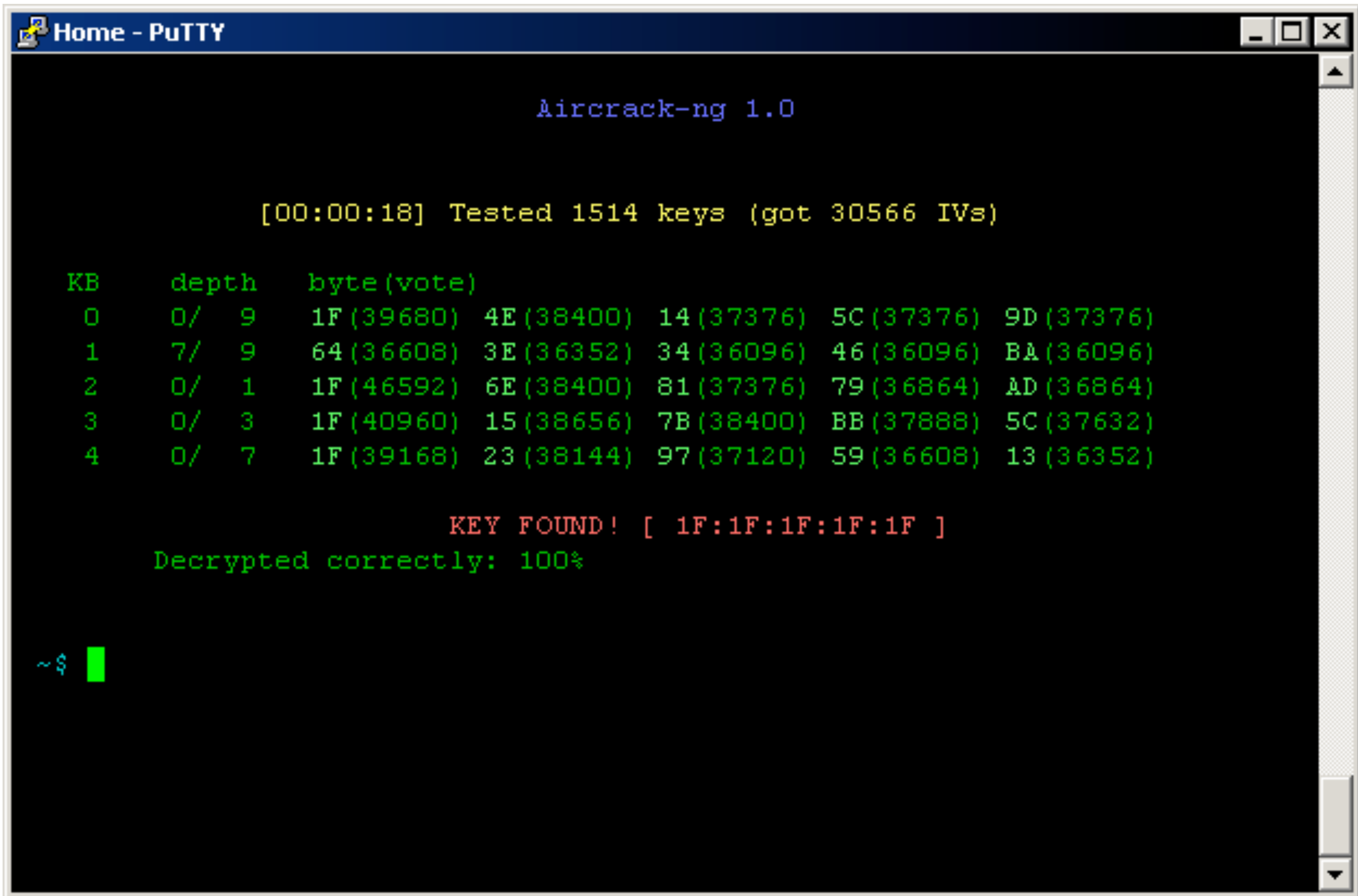
Wi-Fi Protected Access (WPA)

- Encrypting Traffic (2 confidentiality protocols):
  - **AES:**
    - supported in newer WPA2 protocol
    - runs AES in stream-cipher like way (e.g., using something similar to counter mode)

Attacks against WPA

- WPA is a lot stronger than WEP
- Most attacks rely on weak passwords
  - user-supplied keys are either entered as 256-bit string (64 hex digits) or as password
  - password is hashed to produce key using 4096 iterations of HMAC-SHA1 with SSID of AP as salt
  - there exists dictionaries of pre-hashed keys for most popular SSIDs (“linksys”, “redsox”, etc.)

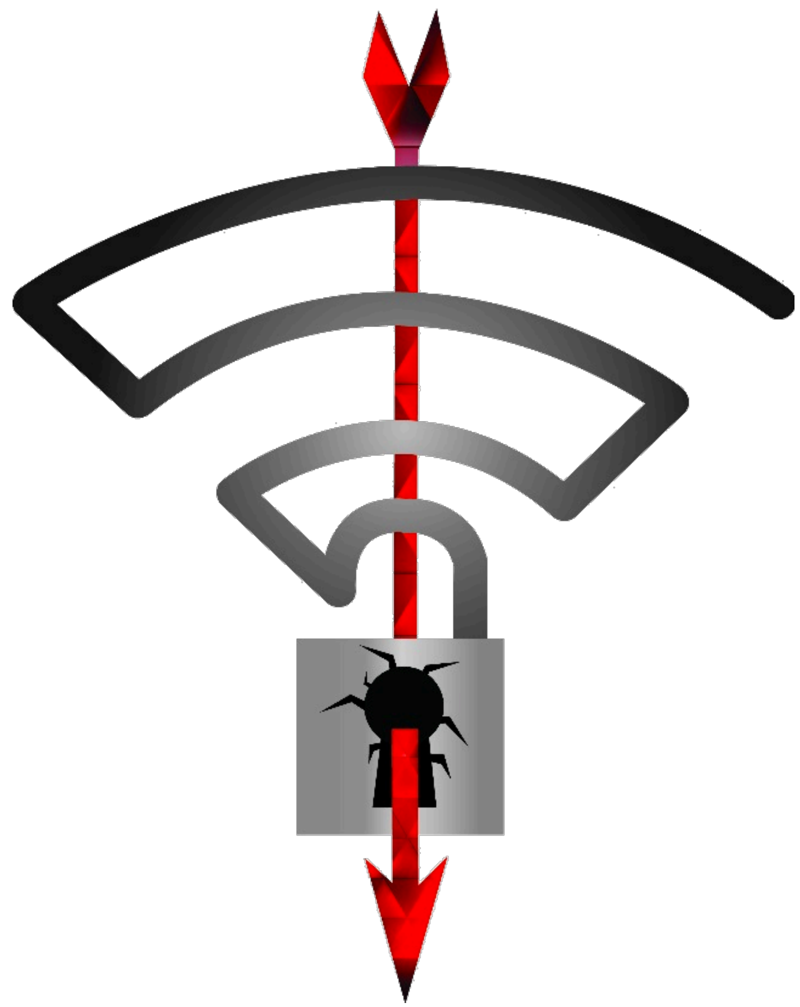
Plenty of tools available
(usually exploit RC4 weakness)



airbase-ng

```
1. sh
AIRBASE-NG(1)                                                              AIRBASE-NG(1)



NAME
       airbase-ng  -  multi-purpose  tool  aimed at attacking clients as opposed to the Access
       Point (AP) itself

SYNOPSIS
       airbase-ng [options] <interface name>

DESCRIPTION
       airbase-ng is multi-purpose tool aimed at attacking clients as opposed  to  the  Access
       Point  (AP)  itself.  Since  it is so versatile and flexible, summarizing it is a chal-
       lenge. Here are some of the feature highlights:
       - Implements the Caffe Latte WEP client attack
       - Implements the Hirte WEP client attack
       - Ability to cause the WPA/WPA2 handshake to be captured
       - Ability to act as an ad-hoc Access Point
       - Ability to act as a full Access Point
       - Ability to filter by SSID or client MAC addresses
       - Ability to manipulate and resend packets
       - Ability to encrypt sent packets and decrypt received packets

       The main idea is of the implementation is that it should encourage clients to associate
       with the fake AP, not prevent them from accessing the real AP.

       A  tap  interface  (atX) is created when airbase-ng is run. This can be used to receive
       decrypted packets or to send encrypted packets.

       As real clients will most probably send probe requests for common/configured  networks,
:
```

Key Reinstallation Attack (KRACK)

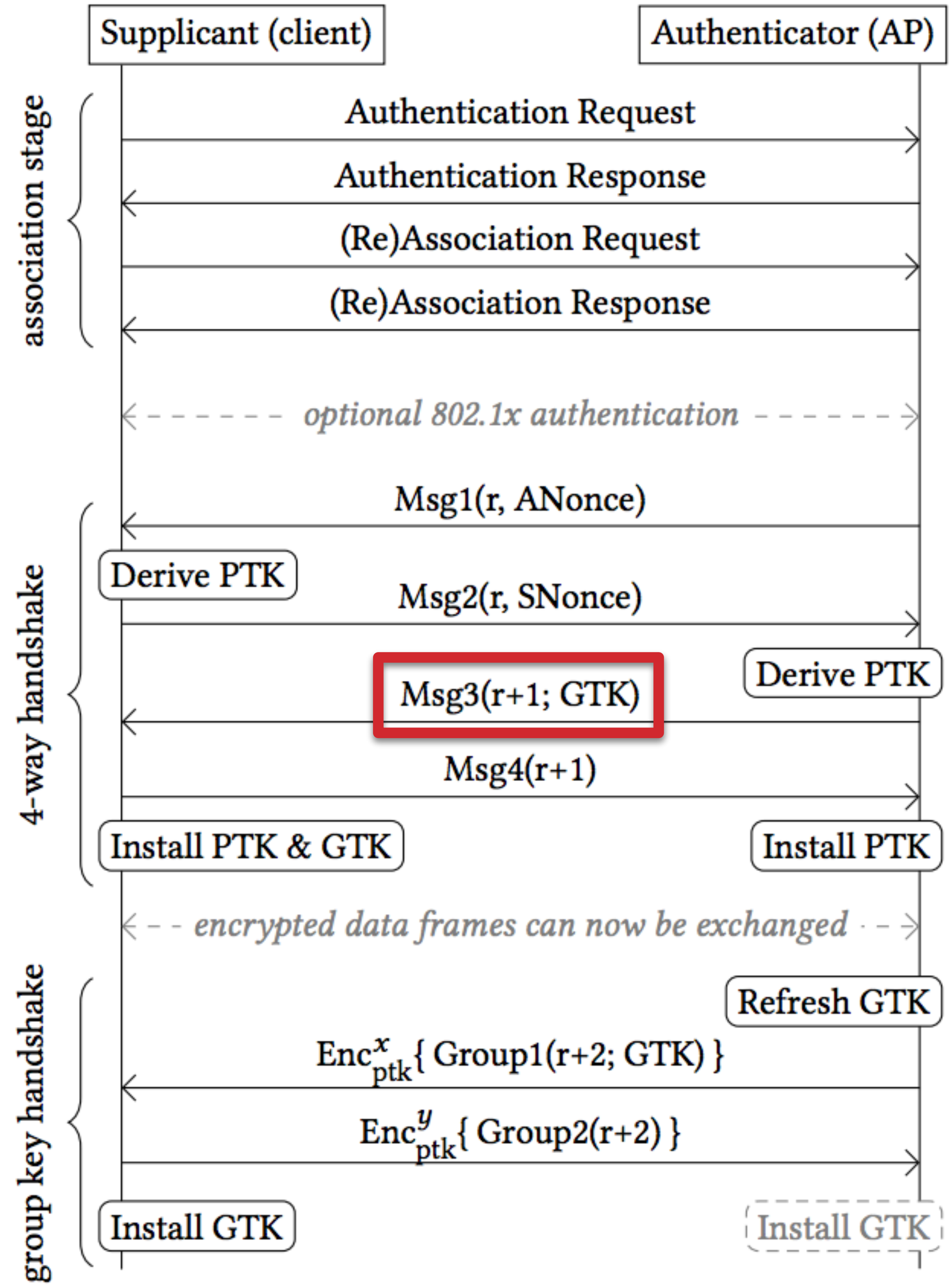


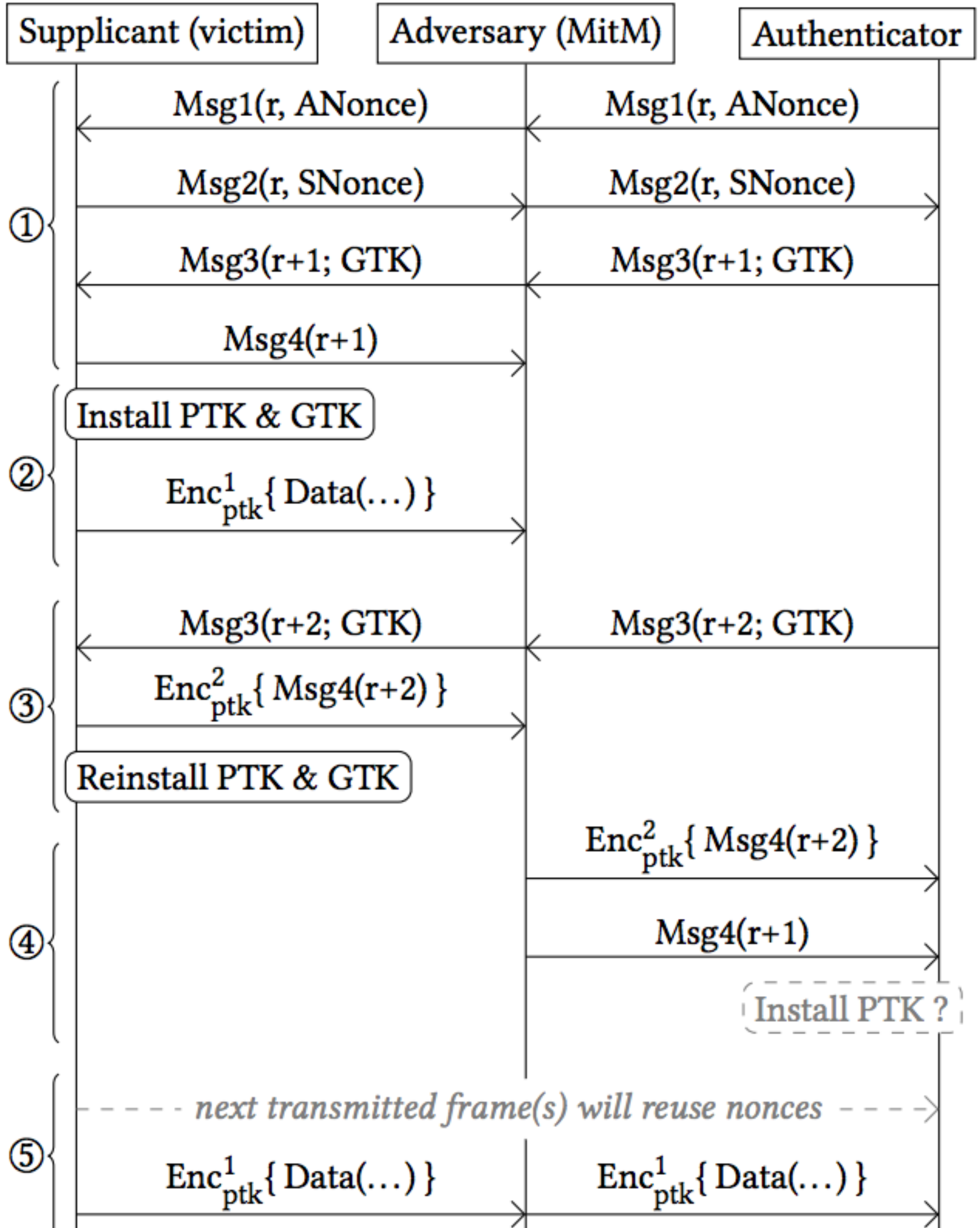
"Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2" - Vanhoef and Piesssens. CCS '17
https://papers.mathyvanhoef.com/ccs2017.pdf

WPA Authentication

Pairwise Transit Key (PTK) = f(PSK, ANonce, SNonce, AP MAC address, STA MAC address)

Pre-Shared Key (PSK)

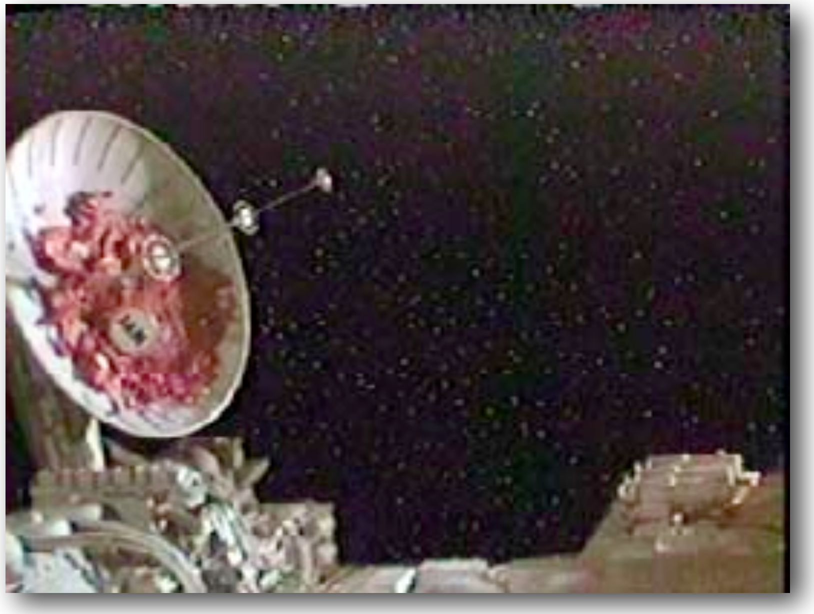




Supplicant (victim)
Adversary (MitM)
Authenticator
①
Msg1(r, ANonce)
Msg1(r, ANonce)
Msg2(r, SNonce)
Msg2(r, SNonce)
Msg3(r+1; GTK)
Msg3(r+1; GTK)
Msg4(r+1)
②
Install PTK & GTK
Enc^1_{ptk} { Data(...) }
③
Msg3(r+2; GTK)
Msg3(r+2; GTK)
Enc^2_{ptk} { Msg4(r+2) }
Reinstall PTK & GTK
④
Enc^2_{ptk} { Msg4(r+2) }
Msg4(r+1)
Install PTK ?
⑤
next transmitted frame(s) will reuse nonces
Enc^1_{ptk} { Data(...) }
Enc^1_{ptk} { Data(...) }

Jamming

- Wireless signals are subject to jamming
- **Analog Jamming:** decrease signal-to-noise ratio by flooding with radio waves
  - basic techniques easy to detect -- just listen for jamming signals
  - more advanced techniques leverage features of the communication system (e.g., FM) to undetectably jam
  - standard defenses: spread spectrum, channel hopping
- **Digital Jamming:** exploit multiplexing scheme to consume all channel bandwidth



Summary

- Wireless basics
- Attacks
  - Eavesdropping
  - Wardriving (and others)
  - KRACK
  - Jamming
- Defenses
  - Configuration-based: MAC filtering, SSID hiding
  - Crypto-based: WEP, WPA2