VU Network Security
Lecture 04

IPv6 Security (continued)
Elgamal
Routing Security

Tanja Zseby
TU Wien

WS 2018/19

Student Position

- 10h/Week
- Minimum: € 500,40 brutto (14x/year)
- Student ETIT or Informatics (or similar)
- Knowledge in IP Networks Network Security, data analysis, programming (C/C++, Python)

- See TU Wien Mitteilungsblatt from 18.10.2018
- Apply at sekretariat@nt.tuwien.ac.at

IPsec Usage in IPv6

Recap: IPsec

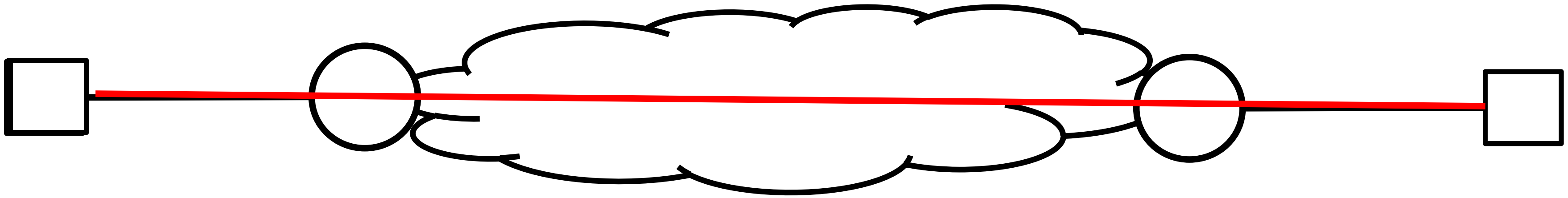
- Security on Network Layer
  - Between Hosts
  - Between Routers
  - Between Router and Hosts

Application
Transport
Network
Data Link
Physical

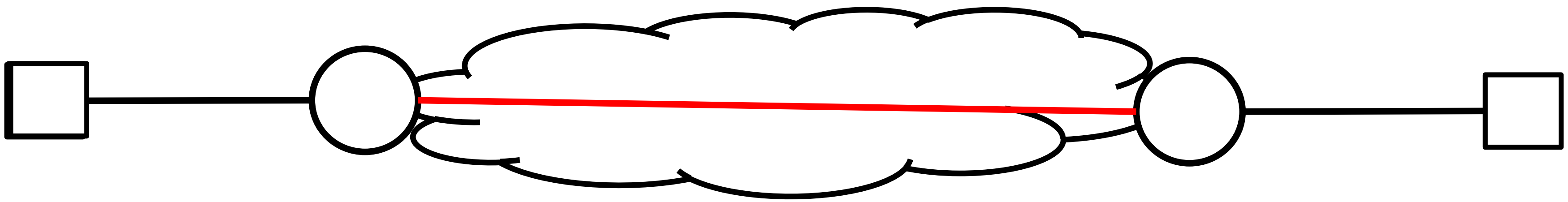
- IP Encapsulating Security Payload (ESP) [RFC 4303]
  - Confidentiality, Integrity
  - MUST be supported
- IP Authentication Header (AH) [RFC 4302]
  - Integrity only
  - MAY be supported

IPsec Modes of Use

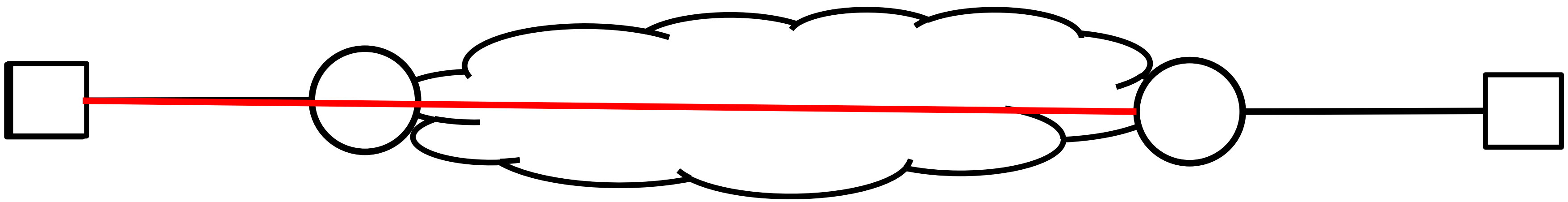
Endpoint-to-Endpoint Transport Mode



Security Gateway to Security Gateway Tunnel Mode



Endpoint to Security Gateway Tunnel Mode



Header Structures (IPv4)

IP Packet

IP Header	Payload (Transport Header, Appl. data)

Transport Mode

IP Header	AH/ESP	Payload

Tunnel Mode

New IP Header	AH/ESP	IP Header	Payload

IPv6 and IPsec

- Described in IPv6 Node Requirements
  - AH and ESP realized as ***Extension Headers***
- RFC 4294 (2006, now obsolete):
  - IPsec MUST be implemented in compliant IPv6 implementation
- RFC6434 (2011):
  - IPsec **SHOULD** be implemented
  - Allow other security solutions:
    - Application-specific solutions
    - Transport layer security
    - Lightweight solutions for devices with limited resources (e.g., sensors)

ESP Extension Header

- ESP protects only fields after the ESP header
- Everything after ESP is encrypted
- Header sequence important
- Router may need to examine
  - hop-by-hop
  - routing
  - (fragmentation)
- Should not be encrypted ➔ placed ***before*** ESP header

IPv6 with ESP (RFC 4303)

- Transport Mode

IPv6 Header	Hop-by-hop, dest*, routing, fragment	ESP	dest *	TCP	Data	ESP Trailer	ESP ICV

← encrypted →

← ICV →

- Destination options extension header(s)
  - Examined by final destination ➔ may be encrypted
  - Put before, after (or both) ESP header
  - Preferably after ESP ➔ protect by ESP

AH Extension Header

- Router may change extension headers
  - Hop-by-hop
  - Routing
- IP header includes mutable fields

- Fields may vary on the path ➔ Integrity check fails

➔ Only include immutable fields in AH calculation

IPv6 with AH (RFC4302)

Some mutable field | immutable fields

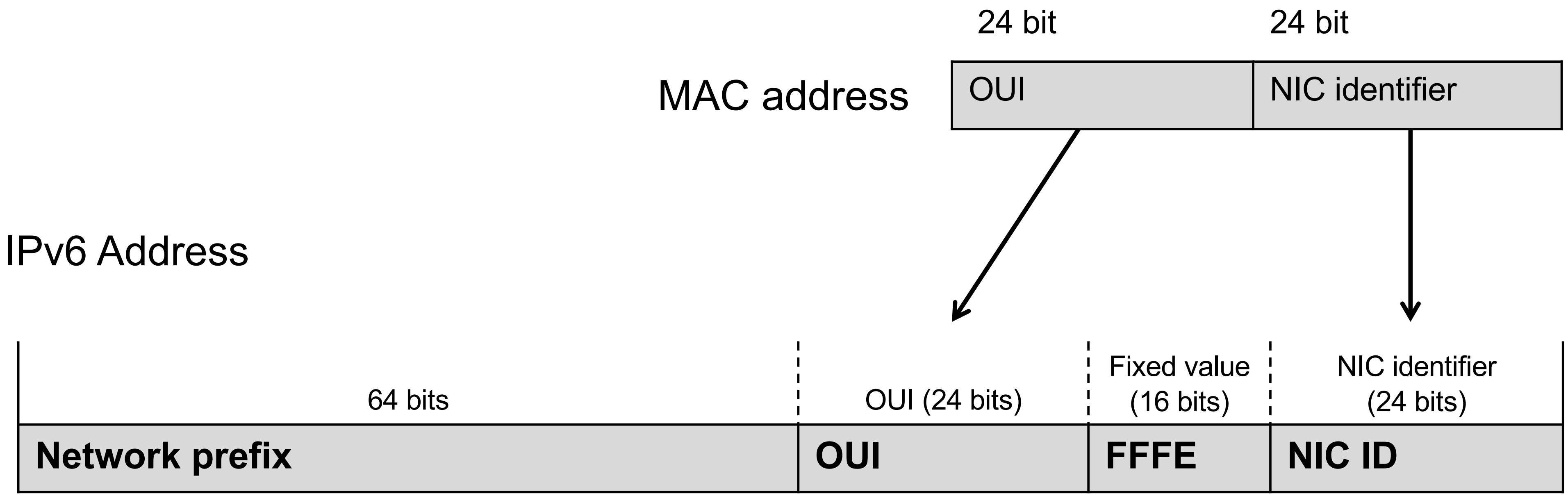
IPv6 Header	Hop-by-hop, dest*, routing, fragment	AH	dest*	TCP	Data

authenticated except mutable fields

- Immutable fields are included in AH calculation
- Mutable fields
  - Are set to zero before ICV calculation
  - If predictable set to predicted value

IPv6 Privacy Issues

Stateless Address Autoconfiguration (SLAAC)



- Generation of IPv6 address based on MAC
  - MAC address is unique and device specific
  - MAC address remains if device moves
  - ➔ IPv6 address can be used to identify device

Privacy Issues with SLAAC

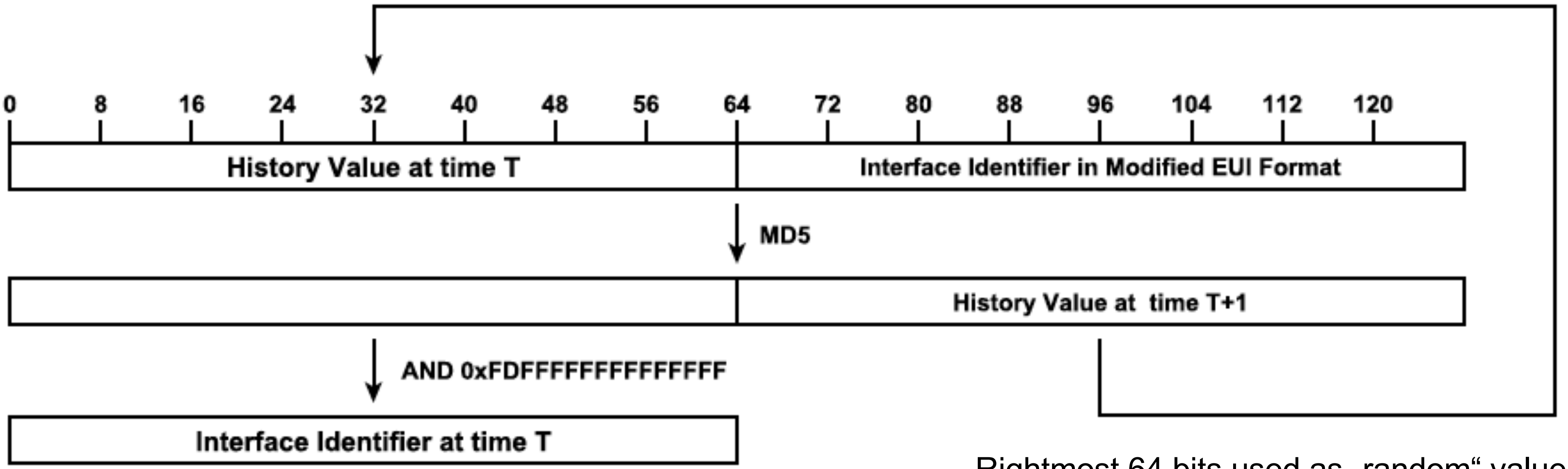
- Advantage
  - Simplifies network administration
  - Useful for fault detection

- Problem: Privacy
  - User profiles based on traffic observation
  - Tracking of movement of devices

Privacy Extension for SLAAC [RFC4941]

- Purpose: generation of temporary addresses
  - Addresses change over time
  - Based on interface identifier and random number
- Generate randomized interface identifier
  - MD5 hash over interface identifier and random number
- Use randomized interface identifier to generate temporary addresses
  - Change addresses from time to time
  - Generate new randomized interface identifier from time to time
- But:
  - Obstructs network analysis
  - Hosts with DNS names can still be identified

Privacy Extension for SLAAC [RFC4941]



Rightmost 64 bits used as „random" value for generation of next address

Leftmost 64 bits form IID for address

Source: J. Ullrich and E. Weippl, "Privacy is Not an Option: Attacking the IPv6 Privacy Extension," in International Symposium on Research in Attacks, Intrusions, and Defenses (RAID), 2015.

Further IPv6 Security Considerations

Extension Headers

"IPv6 nodes must accept and **attempt to process extension headers** in any order and occurring **any number of times** in the same packet, except for the Hop-by-Hop Options header." [RFC 2460]

- Security devices need to parse transport headers
  - ➔ need to parse all headers ➔ potential DoS
- Extension headers may repeat ➔ many headers
- IPv6 encapsulated in IPv6 ➔ even more headers
- Hop-by-hop extension header
  - has several hop-by-hop options
  - any option can appear multiple times
  - Attacker can use inconsistent, invalid options
  - ➔ can cause ICMPv6 error floods

Source: A. Choudhary, "In-depth analysis of IPv6 security posture," *Collaborate Com, 2009*

IPv6 Firewall Configuration

- ICMPv6
  - ICMPv4 usually blocked by firewalls
  - ICMPv6 needed (MTU discovery, autoconfig)
  - Cannot be blocked entirely
  - ➔ recommended filter settings in RFC4890
- Prevent IPv6 extension header attacks
  - ➔ detect unusual header chains, untypical nesting
- Prevent IPv6 fragmentation attacks
  - IPv6 Routers are protected, but
  - Security devices need to parse headers, i.e. reassemble packets
  - ➔ detect unusual amount of fragments

Further IPv6 Security Problems

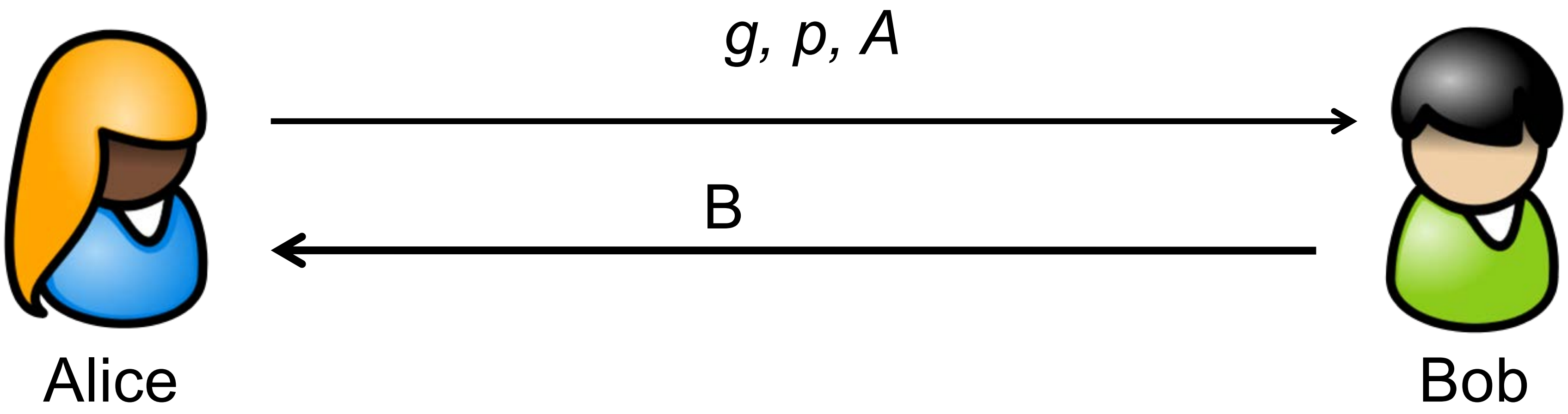
- New technology ➔ new threats
- New implementations ➔ undetected bugs
  - Potential for zero day events
- Lack of experience
  - Configuration errors
  - Tools need to be adapted
  - New traffic profiles ➔ influences detection of anomalies
  - Lack of experts
- IPv4/v6 coexistence
  - Security policies in joint IPv4/IPv6 environments
  - Malware spreading from v4 to v6 or vice versa

Elgamal

Recap: Diffie-Hellman

Select a prime number p
Select a generator g
Generate random number a (secret)
Calculate $A=g^a \bmod p$

Generate random number b (secret)
Calculate $B=g^b \bmod p$



Calculate
$K_{AB}=B^a \bmod p$

- ➔Alice can calculate $\mathbf{K_{AB}}$
- ➔Bob can calculate $\mathbf{K_{AB}}$
- ➔Alice and Bob have a shared secret key

Calculate
$K_{AB}=A^b \bmod p$

Elgamal Encryption

- 1984 by Taher Elgamal

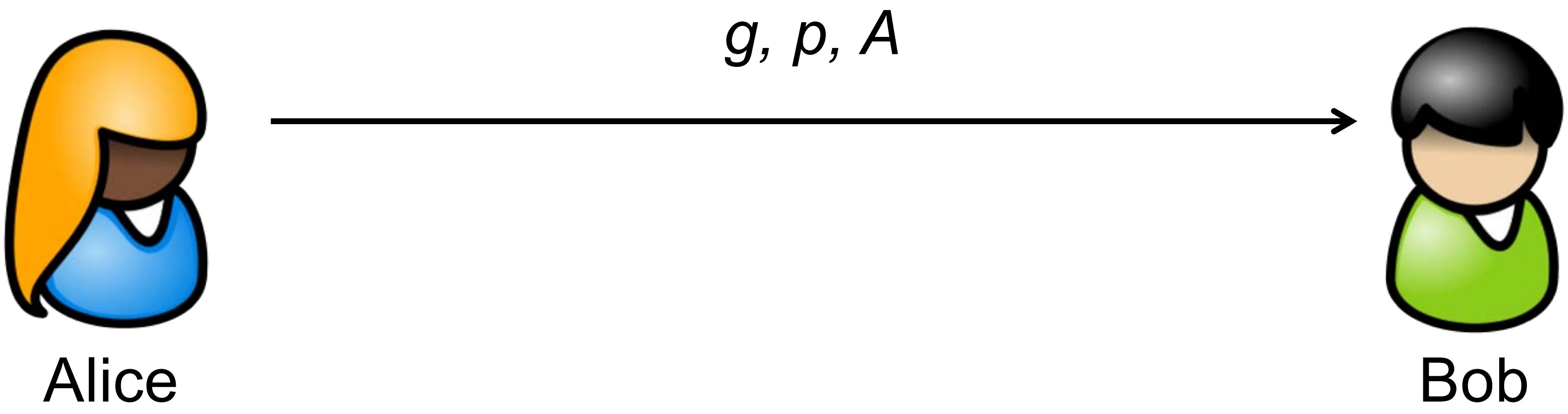
- Uses discrete logarithm
  - Like Diffie Hellman key exchange
  - Prime p
  - Generator g

$$A = g^a \, mod \, p$$

Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," IEEE Transactions on Information Theory, vol. 31, no. 4, Jul. 1985. (earlier in CRYPTO 1984)

Elgamal Encryption

Select a prime number p
Select a generator g
Generate random number a (secret)
Calculate $A=g^a \bmod p$



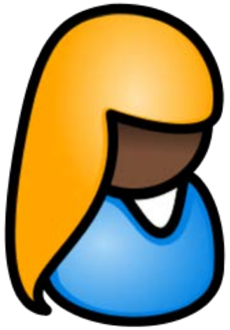
Elgamal Encryption

Generate random number k (secret)
Calculate ciphertext:

$c_1 = g^k \bmod p$

$c_2 = A^k \, m \bmod p$

a



Ciphertext consists of c_1 and c_2
➔ Twice as large as message

c_1, c_2 ←

g, p, A

Alice

Bob

$$m = c_1^{-a} \cdot c_2 \;\; mod\, p$$

$$c_1^{-a} \cdot c_2 \;\; \equiv g^{-ak} \cdot A^k \cdot m \equiv g^{-ak} \cdot g^{ka} \cdot m \equiv m \qquad mod\, p$$

Elgamal Encryption

- Same k for 2 messages?
- ➔ bad idea
- If one plaintext message m_1 is known ➔ adversary can decrypt other ciphertext

$$c_{1,1} = g^k \bmod p \qquad c_{1,2} = A^k \cdot m_1 \bmod p$$

$$c_{2,1} = g^k \bmod p \qquad c_{2,2} = A^k \cdot m_2 \bmod p$$

$$\frac{c_{1,2}}{c_{2,2}} \equiv \frac{m_1}{m_2} \quad \bmod p$$

- ➔ need to change k after each message

Elgamal Encryption

- If new k for each message
  - Ciphertext for same message looks different

$$c_{1,2} = A^{k_1} \cdot m_1 \mod p$$
$$c_{2,2} = A^{k_2} \cdot m_1 \mod p$$

- Secure against **chosen plaintext attacks**
  - Adversary cannot compare pre-calculated ciphertexts for different messages with intercepted ciphertext

$$m_1 = \text{"}YES\text{"}$$
$$c_1 = ?$$

But: Message Manipulation

$c_1 = g^k \; mod \; p \qquad\qquad c_2 = A^k \cdot m \;\; mod \; p$

- Message m_1= 100 $
- Someone on path modifies c_2 :

$$c_2' = 2 \cdot c_2 = 2 \cdot A^k \cdot m \;\; mod \; p$$

- After decryption $m' = 2 \cdot m = 200 \, \$$
- Elgamal is ***malleable***
  - From ciphertext c adversary can generate ciphertext c' that becomes f(m) after decrypting
  - Can be used for ***homomorphic*** encryption
- Use encryption AND Signature
- Use hashed message h(m), not plain message m

Elgamal Signature

- Used in NIST Standard Digital Signature Algorithm

- Message m with $0 \le m \le p-1$
- Public key

$$A = g^a \ mod\, p$$

- Private key *a*

- Create signature in a way that
  - Signing only possible when knowing private key a
  - Verification possible by everyone with public key A

Elgamal Signature

- Chose random k with $0 \leq k \leq p\text{-}1$ *and* $gcd(k, p\text{-}1)=1$ ➔ inverse mod p-1 exist
- Compute r $$r = g^k \ mod\, p$$
- Compute inverse k^{-1} mod p-1
  - Using extended Euclidian Algorithm
- Compute s (if s=0 ➔ chose new k) such that

$$m \equiv ar + ks \ \ mod\, p - 1$$

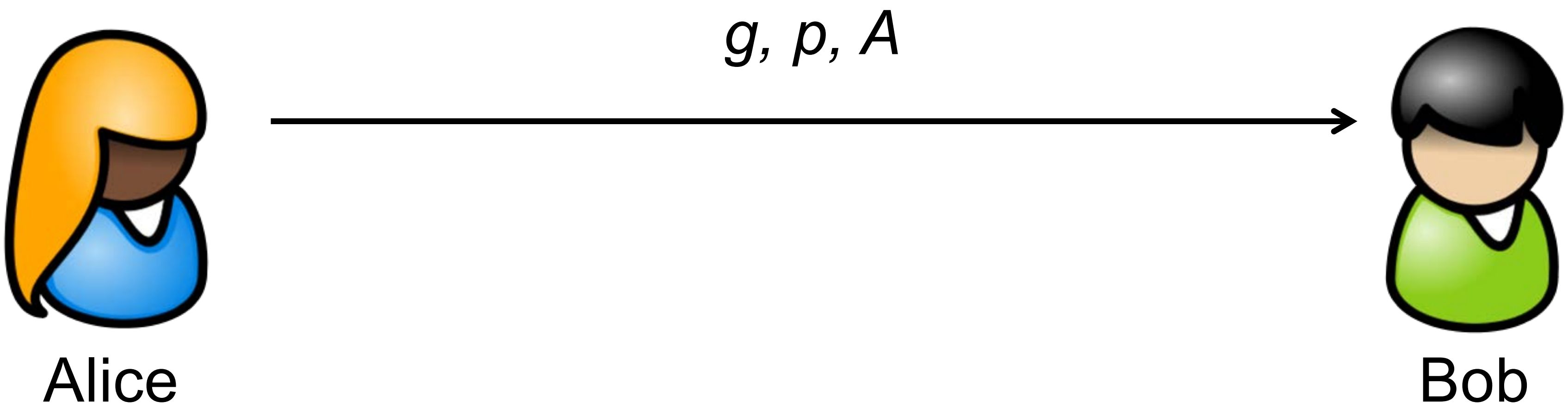
$$s \equiv (m - ar) \cdot k^{-1} \quad mod\, p - 1$$

- Use (r,s) as signature

Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," IEEE Transactions on Information Theory, vol. 31, no. 4, Jul. 1985.

Elgamal Signature

Select a prime number p
Select a generator g
Generate random number a (secret)
Calculate $A=g^a \bmod p$

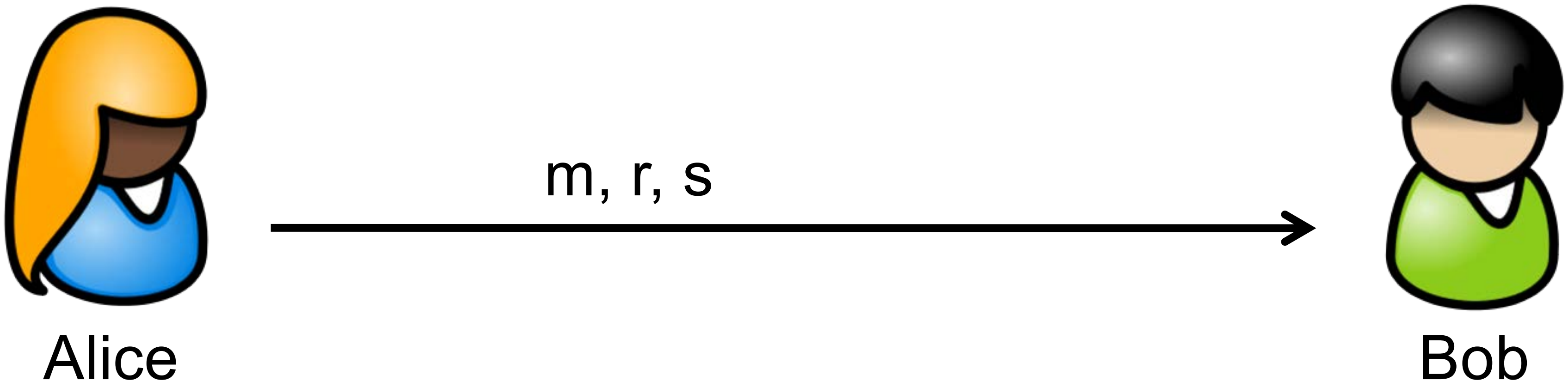


Elgamal Signature

Chose k (secret)
Compute r=g^k mod p
Find s with
m=ar+ks mod p-1

Compute g^m mod p
Compute A^r r^s mod p
Compare if

$$g^m \equiv A^r \cdot r^s \; mod \, p$$



Remark: In practice not m but a hash of m H(m) is used

Verification

- Receiver knows parameters g, p and public key A
- Sender sends m, r, s
- Receiver compares if: $g^m \equiv A^r \cdot r^s \;\; mod\, p$
- Since $m \equiv ar + ks \;\; mod\, p - 1$
- We get: $m + i \cdot (p-1) = ar + ks$

$$A^r \cdot r^s = g^{ar} \cdot g^{ks} = g^{ar+ks} = g^{m+i\cdot(p-1)}$$
$$= g^m \cdot g^{i\cdot(p-1)} = \; g^m \cdot \left(g^i\right)^{p-1}$$

- With Euler $x^{\varphi(n)} \equiv 1 \qquad mod\, n$

$$\left(g^i\right)^{p-1} \equiv 1 \quad mod\, p$$

$$A^r \cdot r^s \equiv \; g^m \quad mod\, p$$

Elgamal Signature

- If same k is used twice

$$m_1 \equiv ar_1 + ks_1 \;\; mod\, p - 1$$

$$m_2 \equiv ar_2 + ks_2 \;\; mod\, p - 1$$

- m_1,m_2 may be known (authentication without secrecy)
  ➔2 equations, 2 unknowns a, k
  ➔ adversary can deduce secret key a

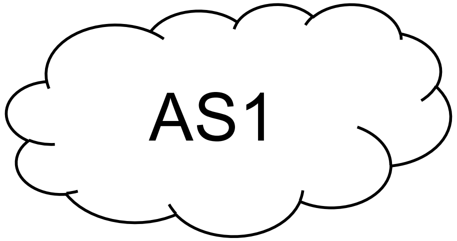
- k should only be used once
  ➔ generate new k for each message

Elgamal

- Elgamal
  - Ciphertext for same message differs (protects against forward search)
  - But: Ciphertext twice as long as plaintext
  - New random number needed for each encryption or signature
  - Needs very good random number generator
- Digital Signature Algorithmus (DSA)
  - US standard for digital signatures
  - Recommended by NIST to be used in Digital Signature Standard (DSS)
  - Variant of the Elgamal signature scheme

Routing Security

Recap: Autonomous System

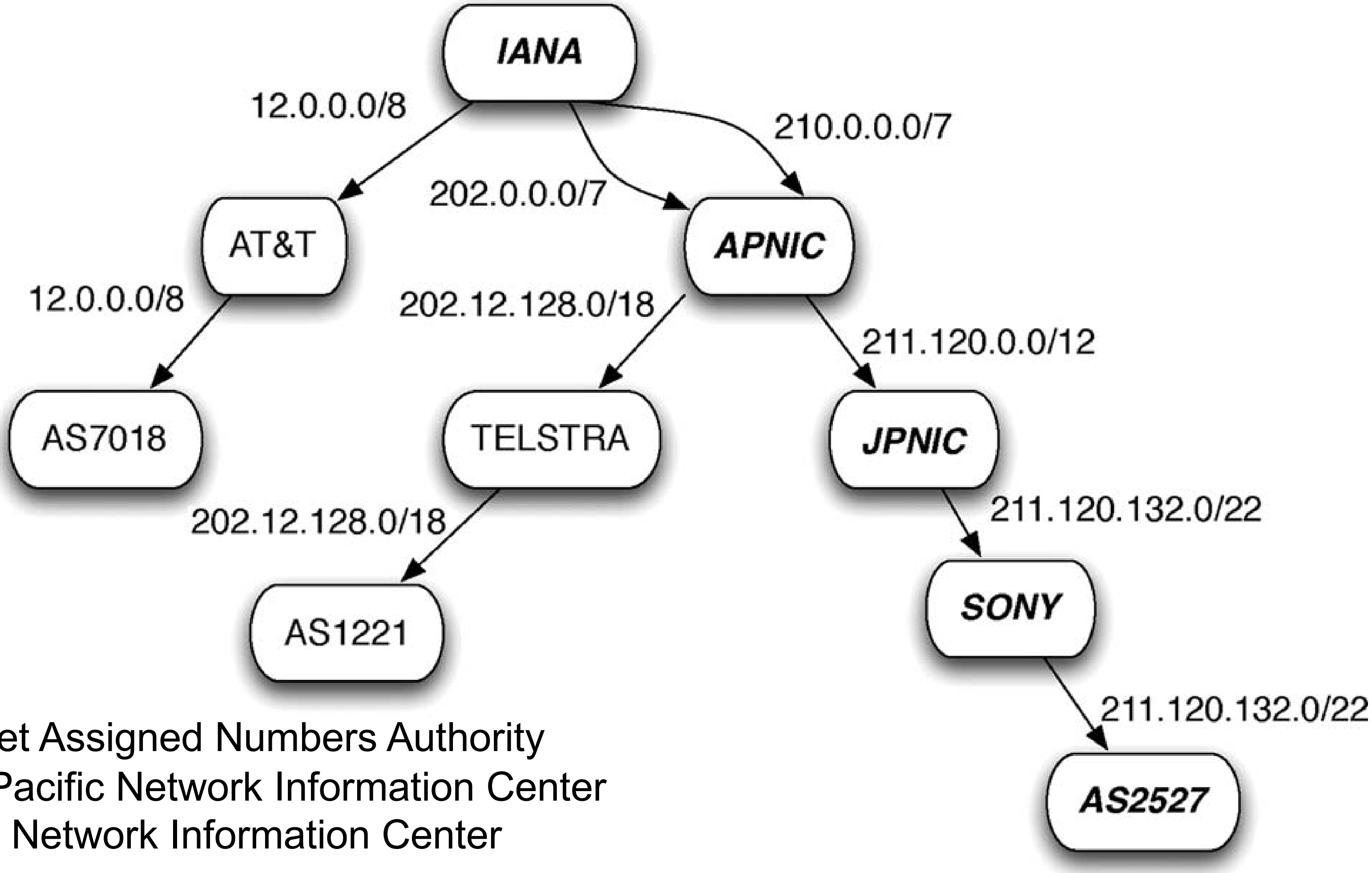


- Collection of networks
- Single administrative control
- Same routing policy

- Unique AS number for each AS
- Intra-AS vs. Inter-AS routing

Address Delegation

- Allocation of Address space and AS numbers



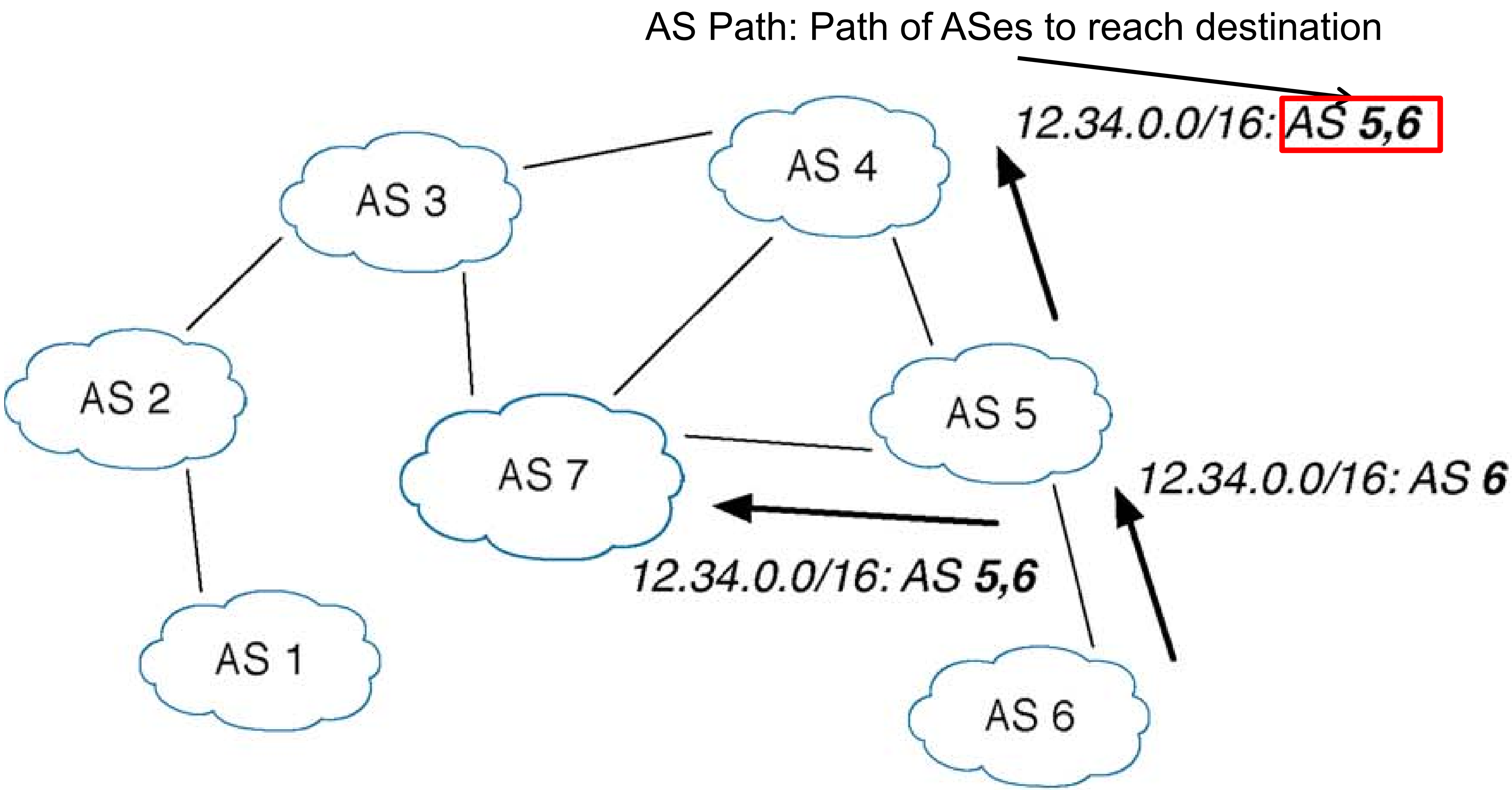
IANA - Internet Assigned Numbers Authority
APNIC- Asia-Pacific Network Information Center
JPNIC- Japan Network Information Center

Source: Butler et al., "A Survey of BGP Security Issues and Solutions," *Proceedings of the IEEE*, vol.98, no.1, Jan. 2010

Recap: Border Gateway Protocol (BGP)

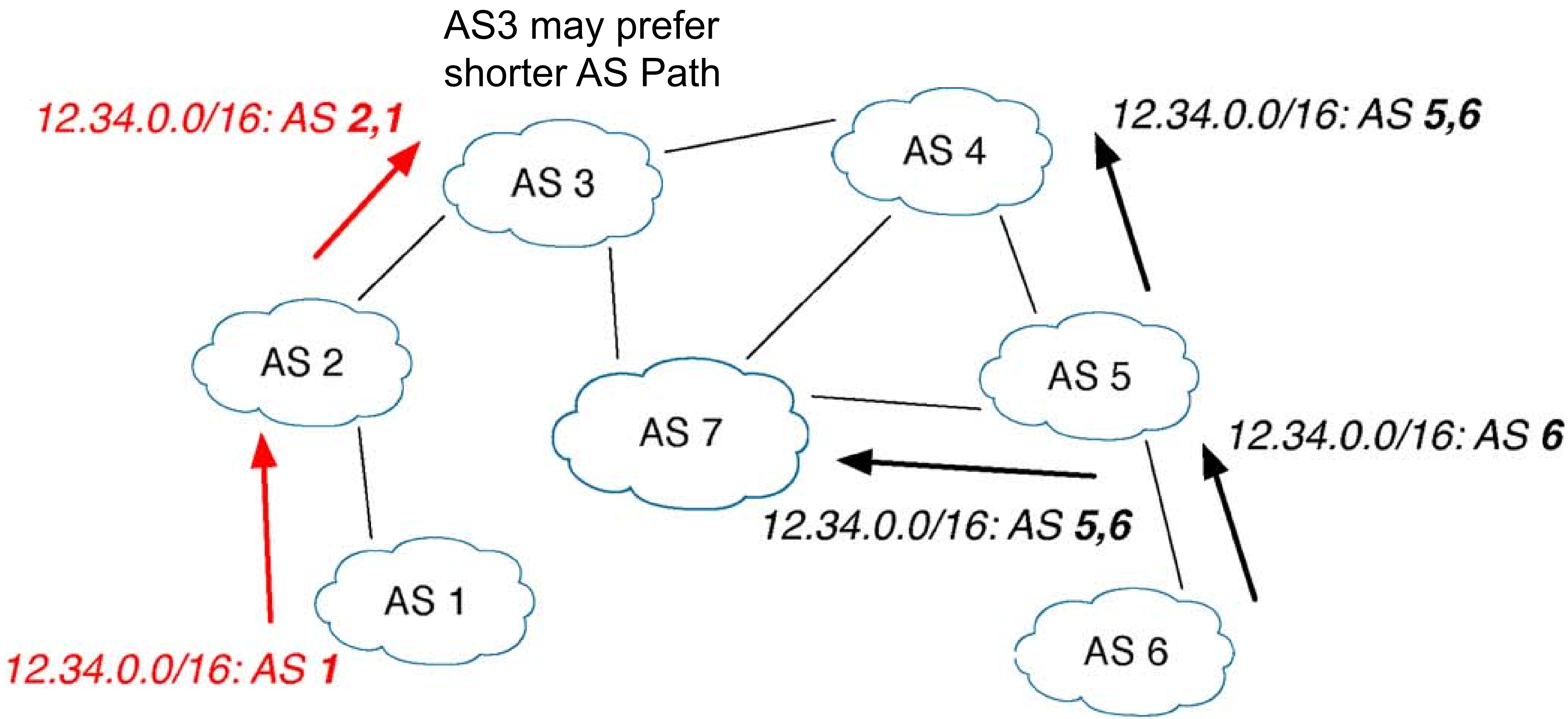
- BGP-4 described in RFC 4271
- Inter-Domain Routing between ASes
- Widely used in Internet
- Path Vector Protocol
  - Routers advertise reachability of neighbors
  - Each router adds own AS to path vector in advertisement
  - ➔ path vector provides path to destination
- Runs over TCP port 179

Recap: Border Gateway Protocol (BGP)



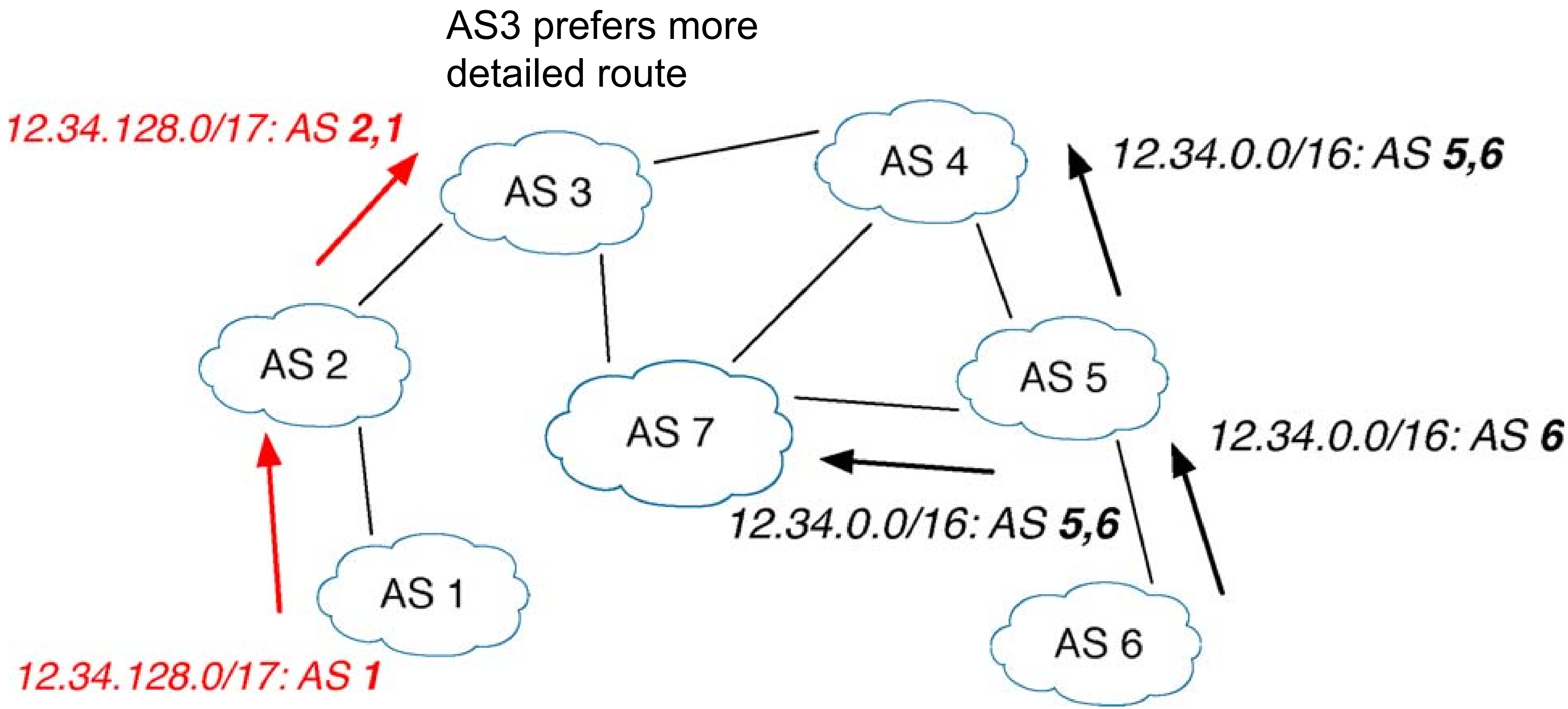
Source: Butler et al., "A Survey of BGP Security Issues and Solutions," *Proceedings of the IEEE*, vol.98, no.1, Jan. 2010

BGP: Malicious Advertisement



Source: Butler et al., "A Survey of BGP Security Issues and Solutions," *Proceedings of the IEEE*, vol.98, no.1, Jan. 2010

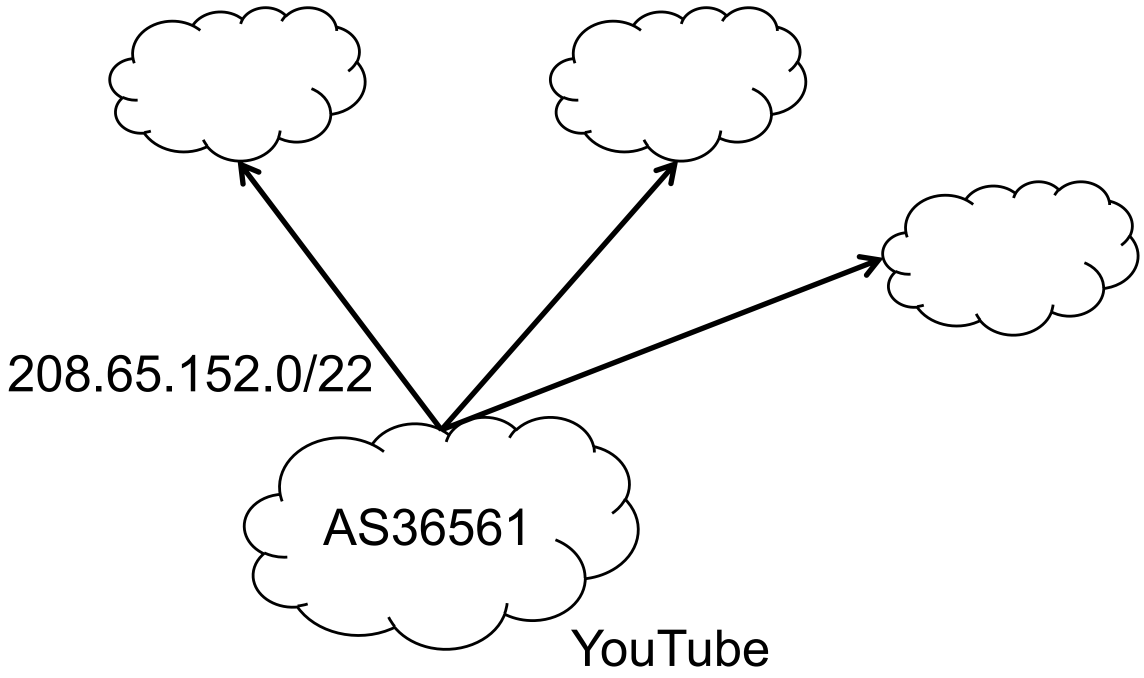
Longest Prefix Match



Source: Butler et al., "A Survey of BGP Security Issues and Solutions," *Proceedings of the IEEE*, vol.98, no.1, Jan. 2010

Prefix Hijacking DoS on YouTube

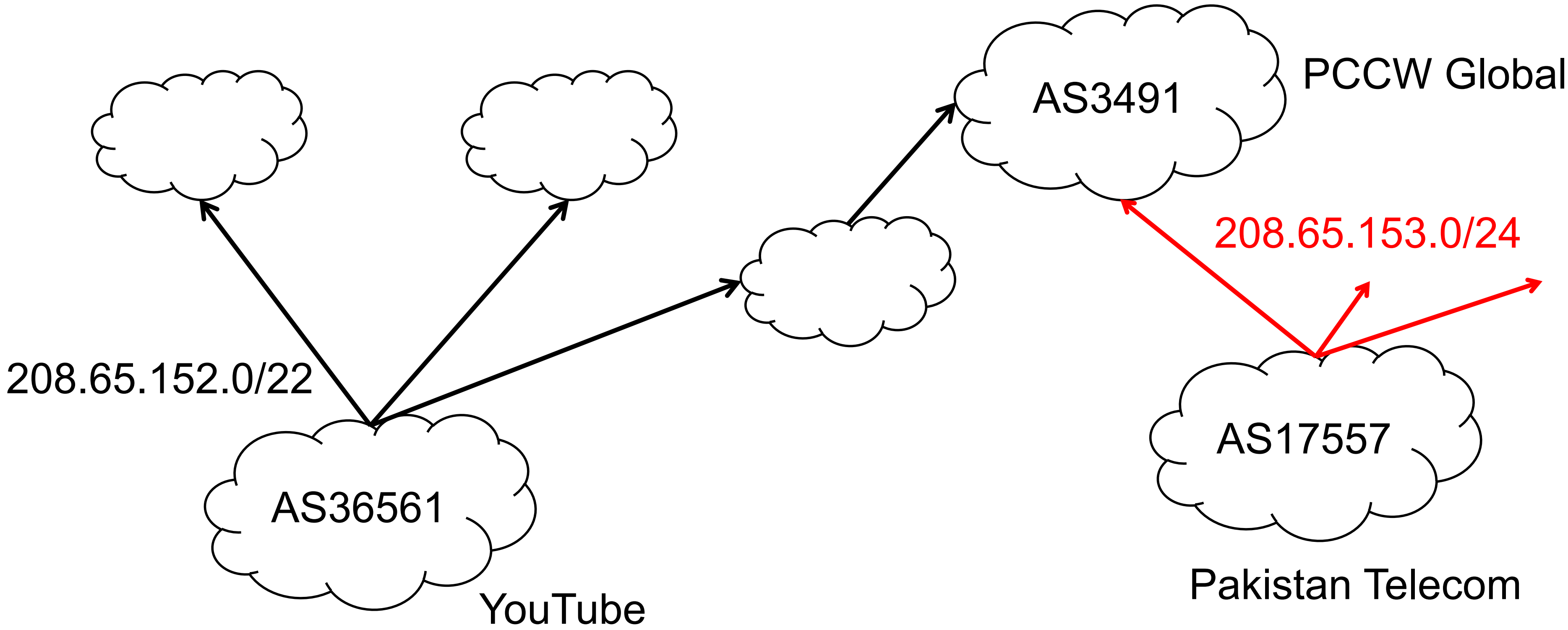
- youtube.com IP addresses:
  - 208.65.153.238, 208.65.153.251, 208.65.153.253
  - AS36561 (YouTube) announces 208.65.152.0/22



Source: YouTube Hijacking: A RIPE NCC RIS case study
http://www.ripe.net/internet-coordination/news/industry-developments/youtube-hijacking-a-ripe-ncc-ris-case-study

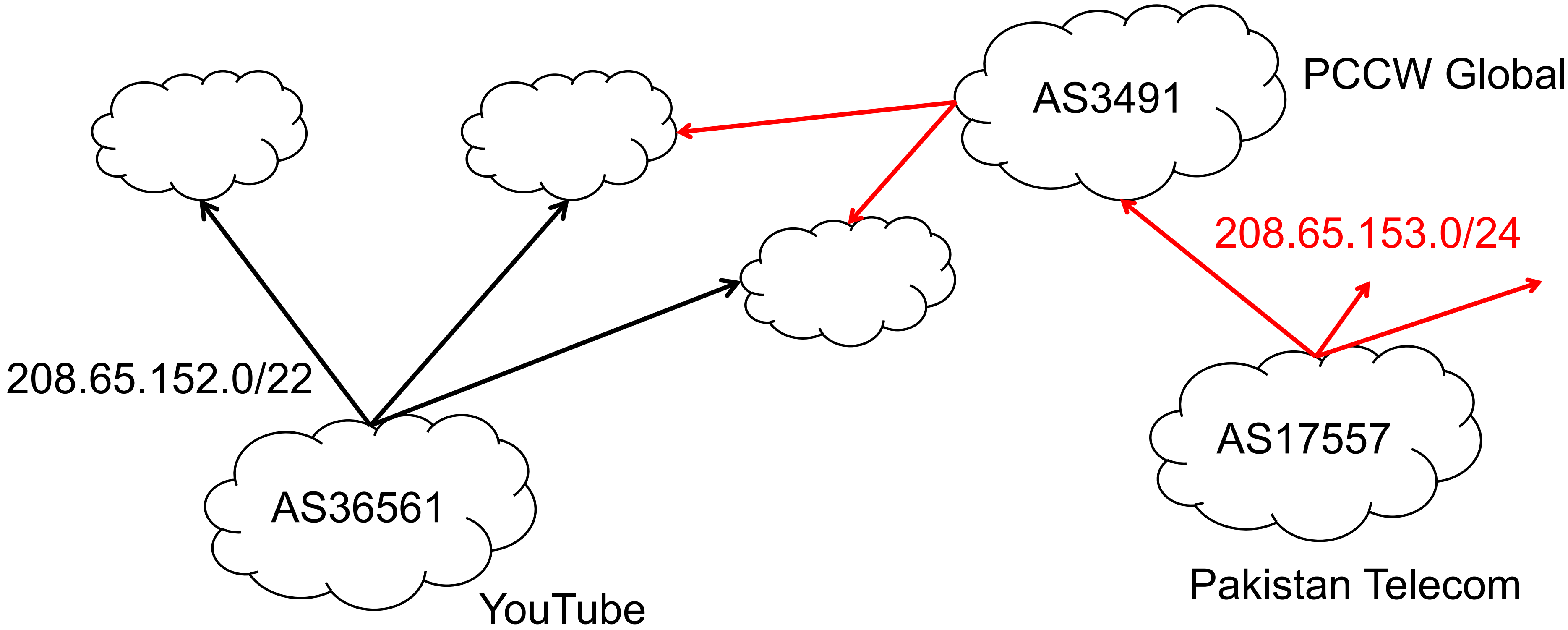
24 February 2008,18:47 (UTC)

- Pakistan Government wants to block YouTube traffic in Pakistan
  - Pakistan Telecom announces 208.65.153.0/24
  - Goal: Redirect YouTube traffic in Pakistan to AS17557

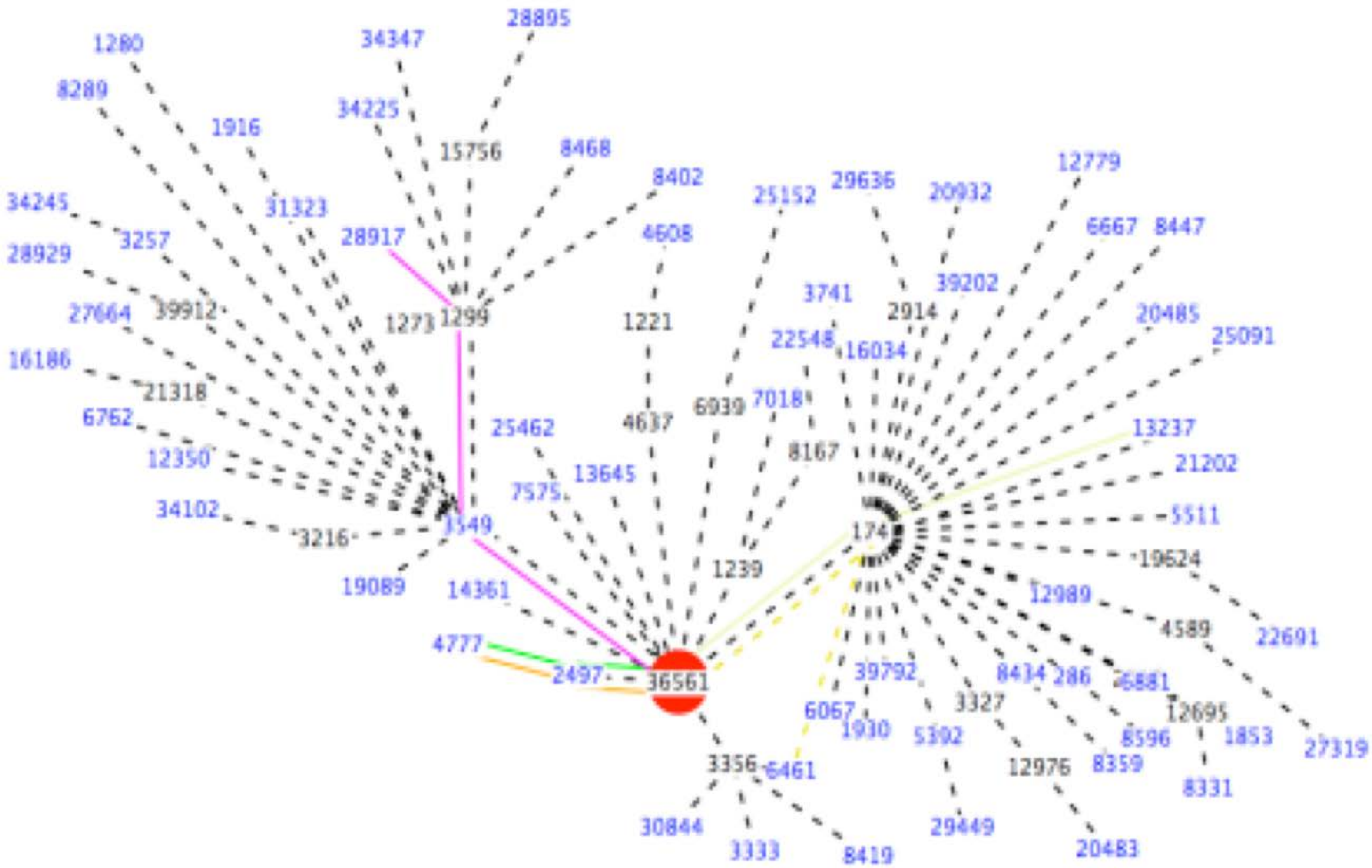


24 February 2008,18:47 (UTC)

- PCCW Global forwards Route
- Routers around the world receive the announcement
- Result: All YouTube traffic redirected to Pakistan

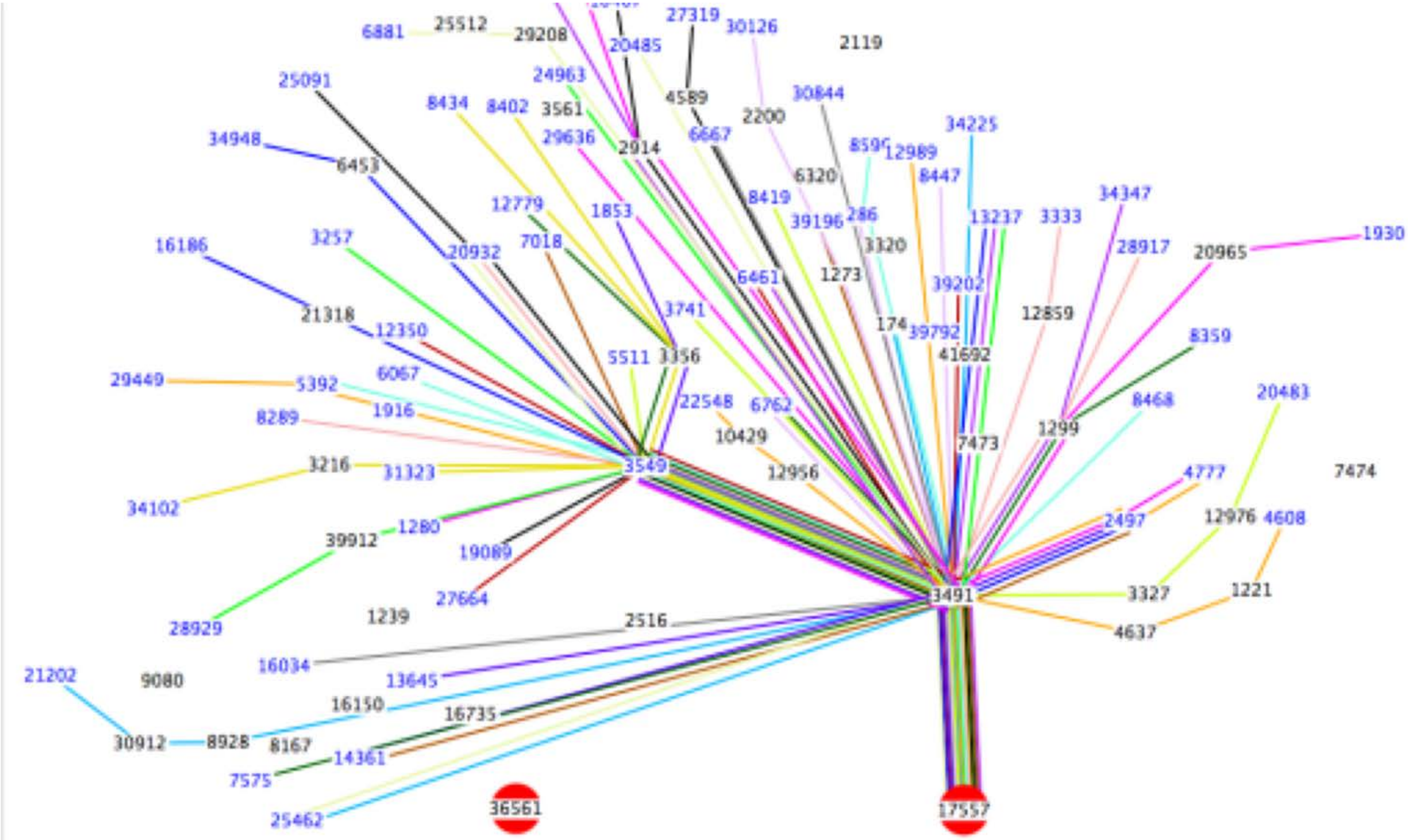


Before Announcement



Source: YouTube Hijacking: A RIPE NCC RIS case study (Tool: BGPlay)
Watch at http://www.youtube.com/watch?v=IzLPKuAOe50

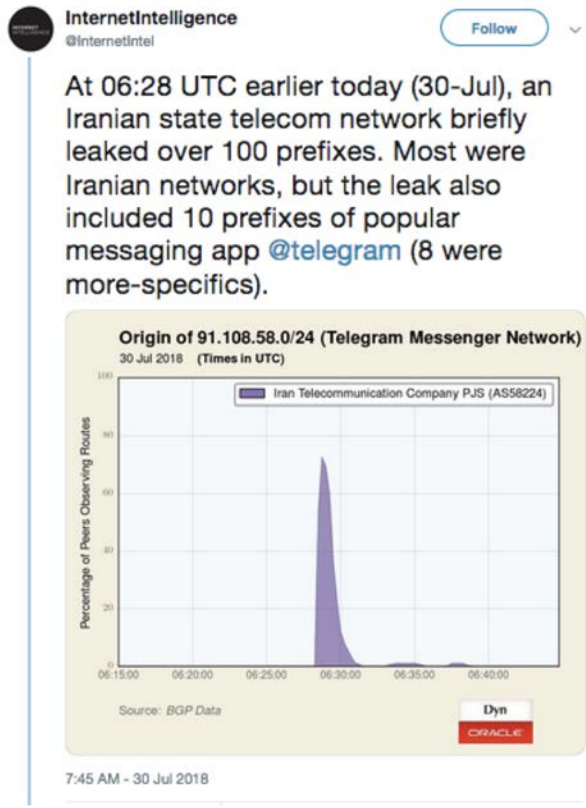
After Announcement



Source: YouTube Hijacking: A RIPE NCC RIS case study

BGP Problem: Prefix Hijacking

- Several similar examples
  - e.g., 30.July 2018
  - Telegram messaging App rerouted via Iran



- Problem
  - BGP routers can announce wrong routes
  - Routers under different administrative control
  - No check for address ownership

Further BGP Problems

- BGP uses TCP
  - Without any security features (no authentication, integrity check, encryption)
  - Vulnerable to TCP attacks (e.g., SYN Flooding)
- No encryption ➔ Eavesdropping on BGP messages
  - Learn routing information and policies
  - Reveal business relationships (e.g., peering)
- No integrity check ➔ Man-in-the-middle
  - Modification of BGP messages
  - Insert wrong or inconsistent information
  - Delete keep-alive messages ➔ kill communication
  - Replay messages (re-assert or withdraw routers)

BGP Security

Some Ideas to Secure BGP

- MD5 Integrity check
  - Message authentication based on keyed MD5
  - Requires shared secret key on routers
  - Partially used today
- Generalized TTL Security Mechanism (GTSM)
  - Protect against remote attacks
  - Routers send BGP packets with TTL=255
  - Receiving router discard packet if TTL<254
  - Partially used today
- Using IPsec
  - Protect communication among routers
  - IPsec used as building block

Some Ideas to Secure BGP

- Filtering suspicious BGP messages
  - Special use addresses (loopback, reserved)
  - Bogons (bogus IP addresses)
    - address blocks, AS numbers with no matching allocation data → List of bogons: http://www.cidr-report.org/as2.0/
  - Small subnets (e.g. /24)
- Limit number of announcements per neighbor
  - Discard if more announcements sent
- Routing registries
  - ASes register their policies, topology
  - → create a global view

IETF Secure Inter-Domain Routing (SIDR) Group

- Objective: reduce vulnerabilities in inter-domain routing
  - ***Challenge 1:*** Is an Autonomous System (AS) *authorized to originate an IP prefix*?
  - ***Challenge 2:*** Is the *AS-Path* in the BGP message the *same as the path* through which the BGP message *traveled*?
- Work on an overall Secure BGP architecture
  - Based on Resource Public Key Infrastructure (RPKI) that represents address allocation hierarchy
- Very active group
  - http://datatracker.ietf.org/wg/sidr/

BGPSEC

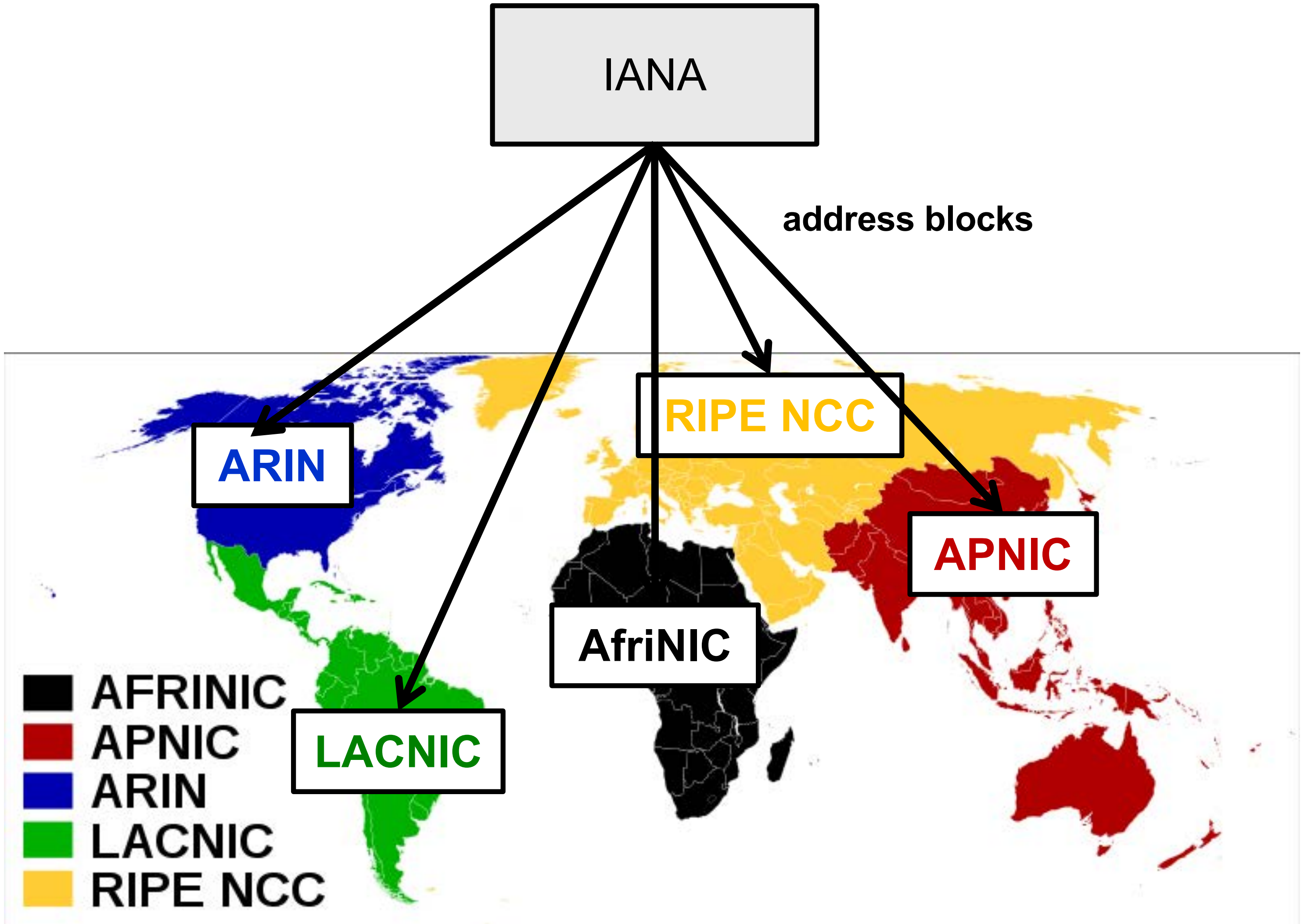
- Recently selected by SIDR
  - Among several proposals
- RFC 8205: BGPsec Protocol Specification (Sept 2017)
- Main idea: Authenticate prefix origins
  - Digital Signature to sign announcements
  - Route Origin Authorization (ROA) Certificates to ensure binding of key to entity
  - Authorize entity to advertise a prefix

Attestations About Addresses

- Goal: Ensure that
  - AS numbers are valid
  - Entities announcing IP addresses and AS numbers are authorized to do so

- Idea: Use address allocation structure
  - Internet Assigned Numbers Authority (IANA) issue certificates to regional registries
  - Certificates grant the right to use a resource (IP addresses, AS numbers)
  - Receiver can validate that originator of (signed) announcement has right to use addresses

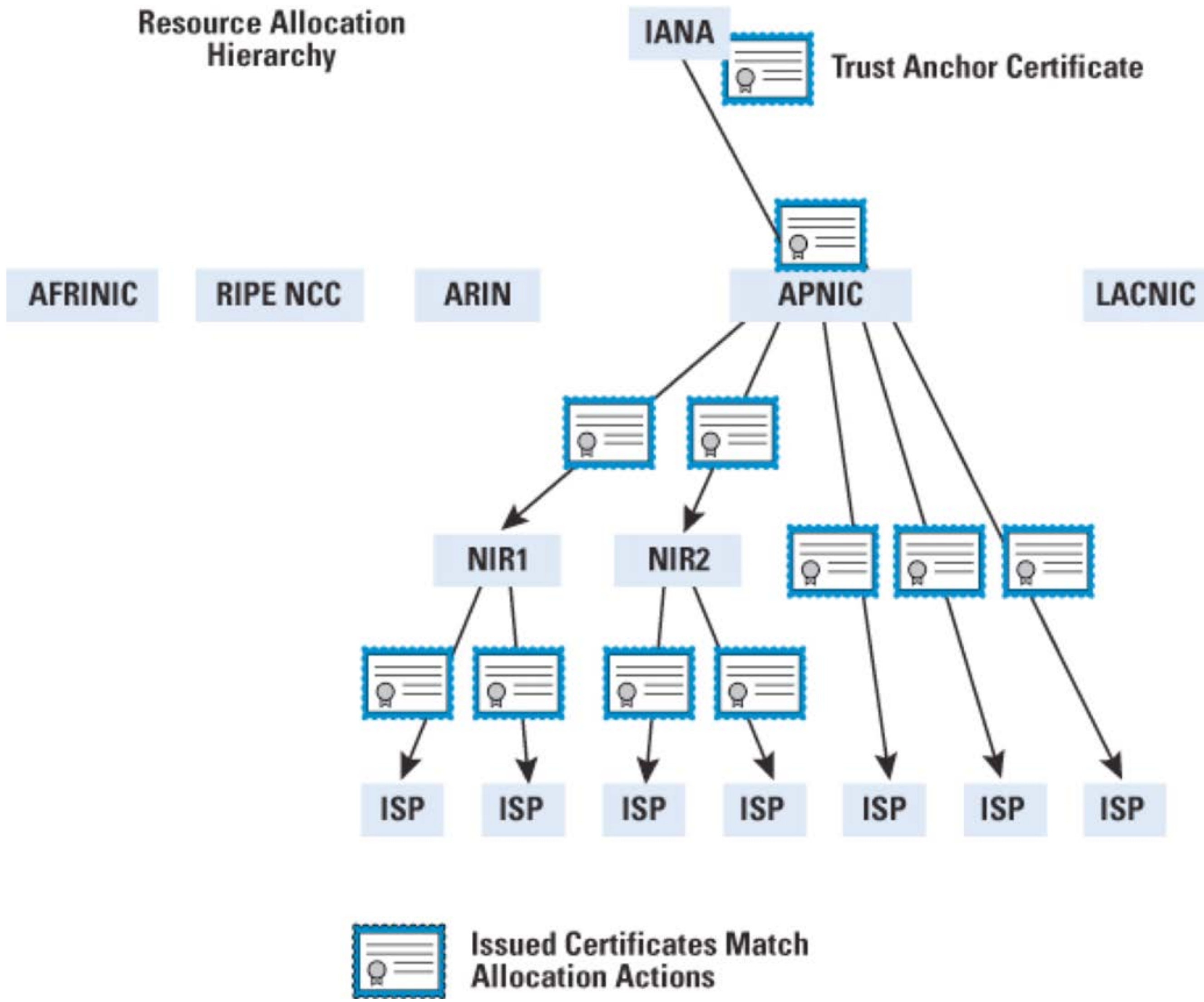
Internet Resource Allocation



Regional Internet Registries (RIRs)

Source: RIR Picture: wikipedia

Resource Public Key Infrastructure (RPKI)



Regional Internet Registries (RIRs)

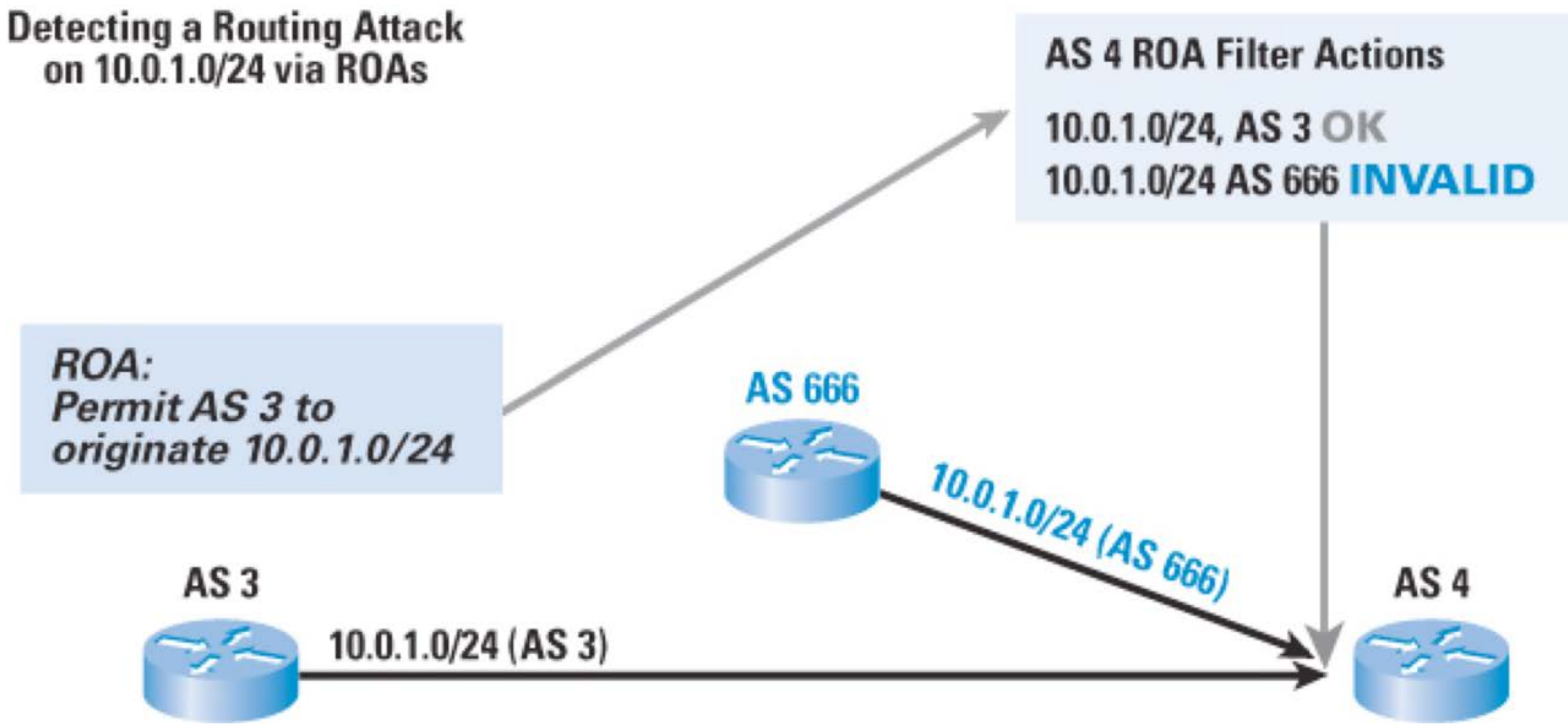
National Internet Registries (NIRs)

Internet Service Providers (ISPs)

Source: Geoff Huston, Randy Bush, Securing BGP, The Internet Protocol Journal, Volume 14, No. 2, 2011

Route Origin Authorization (ROA)

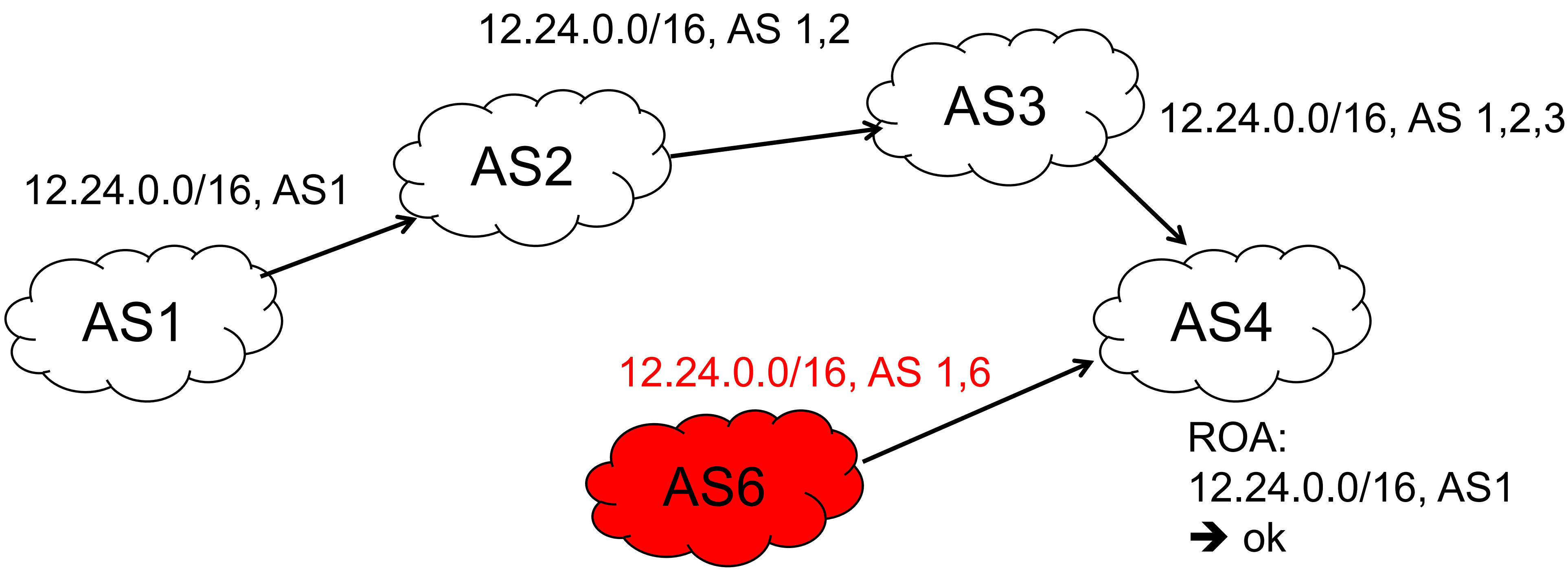
- ROA binds address range to AS number
- Digitally signed by address holder
- Address holder can issue ROAs to others



➔ AS4 can check if AS 3 allowed to origin address prefix

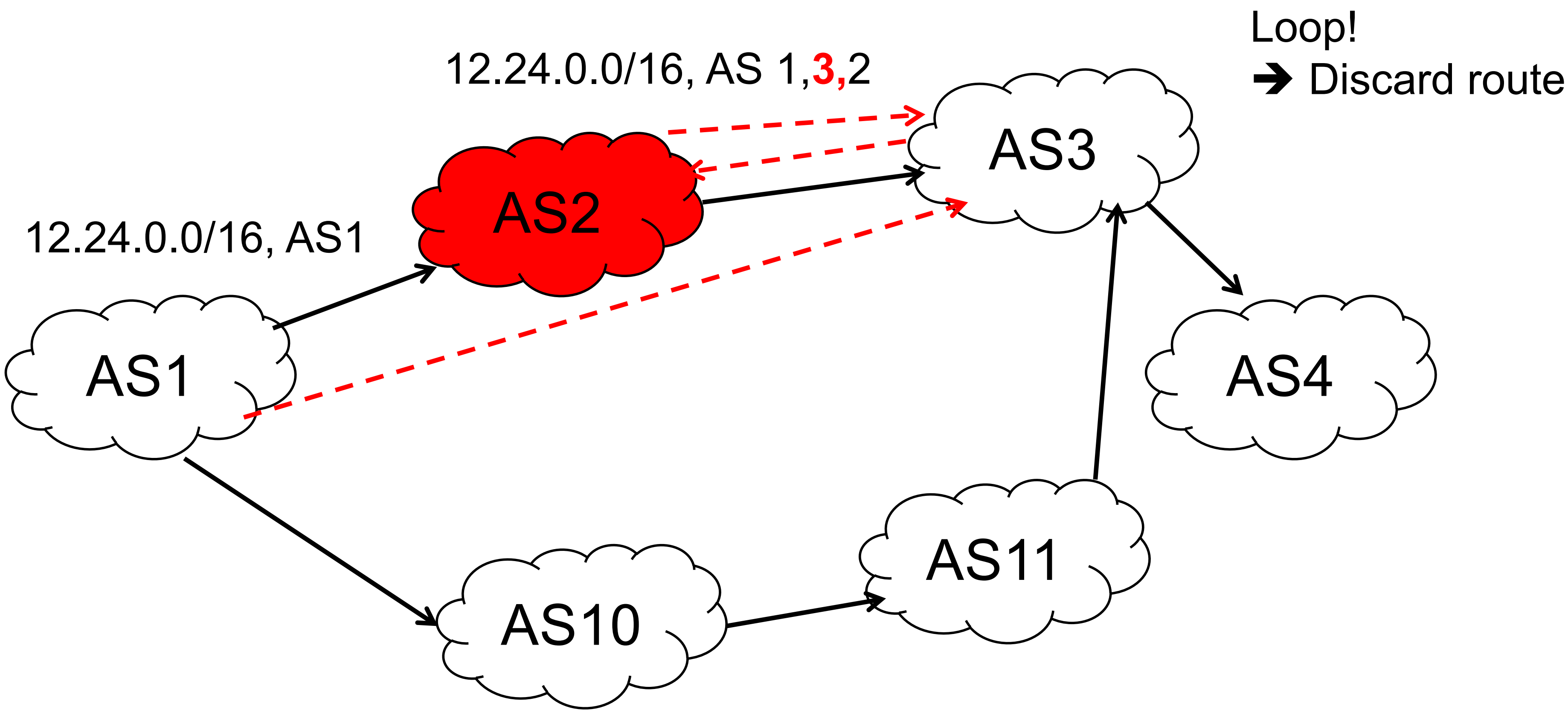
Source: Geoff Huston, Randy Bush, Securing BGP, The Internet Protocol Journal, Volume 14, No. 2, 2011

But: Route Announcements



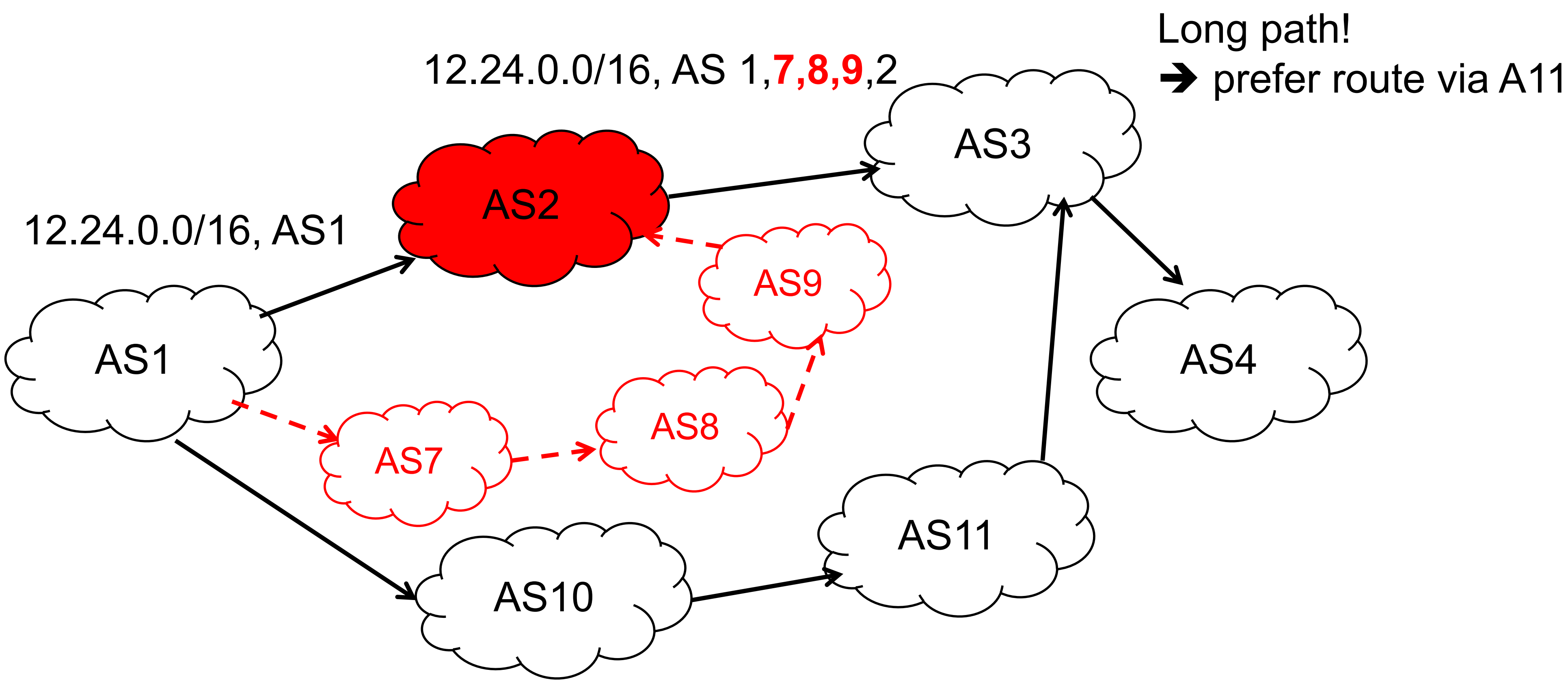
- ROA does not ensure that origin AS in BGP route was indeed the originating AS of this route
- Malicious BGP router may announce **wrong path**

Route Announcements



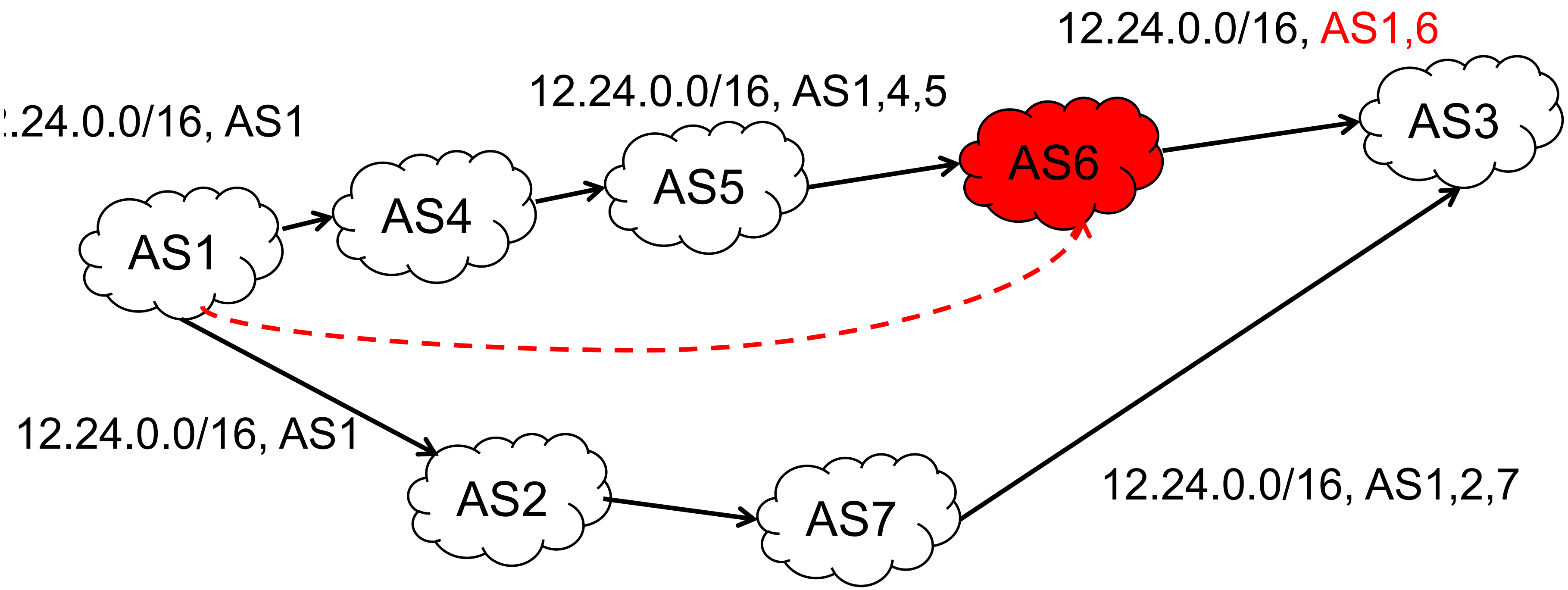
- Malicious BGP router may add receiver AS to path
- Receiver AS detects loop ➔ will not use route

Route Announcements



- Malicious BGP router may add multiple ASes to path
- Long path ➔ Receiver AS will prefer shorter routes

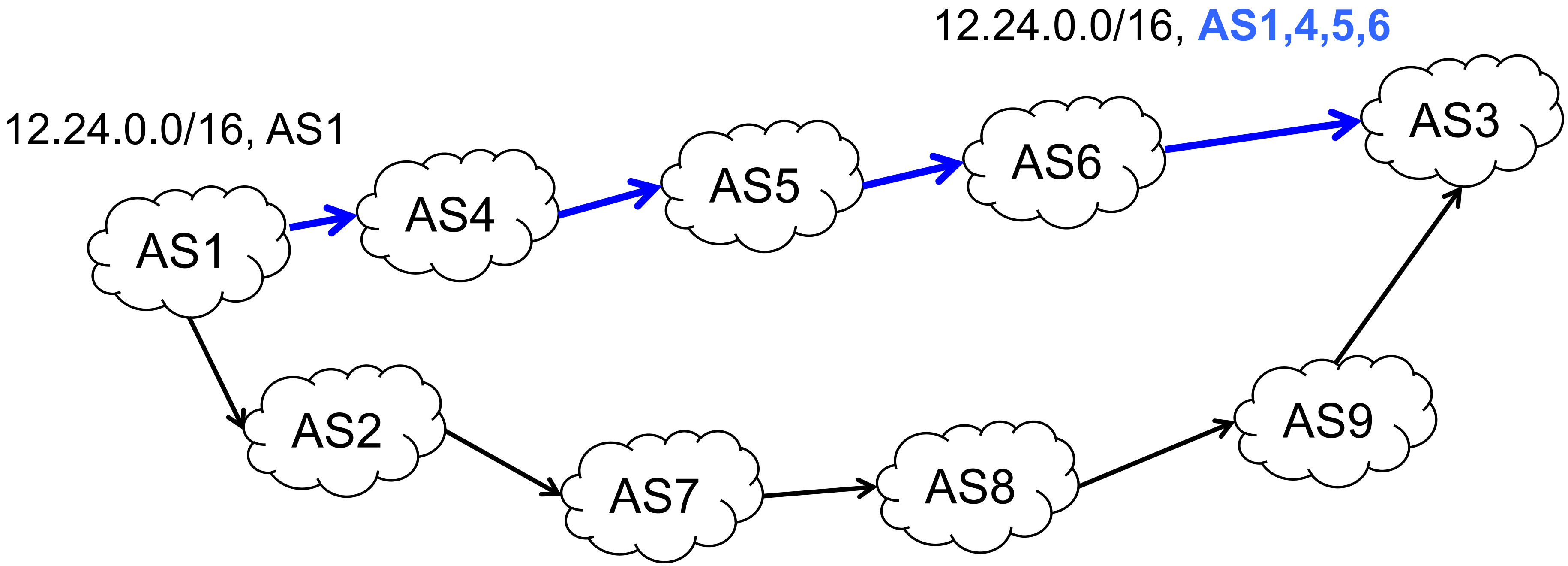
Route Announcements



- Malicious BGP router may remove ASes from path
- Receiver AS will prefer route via malicious AS

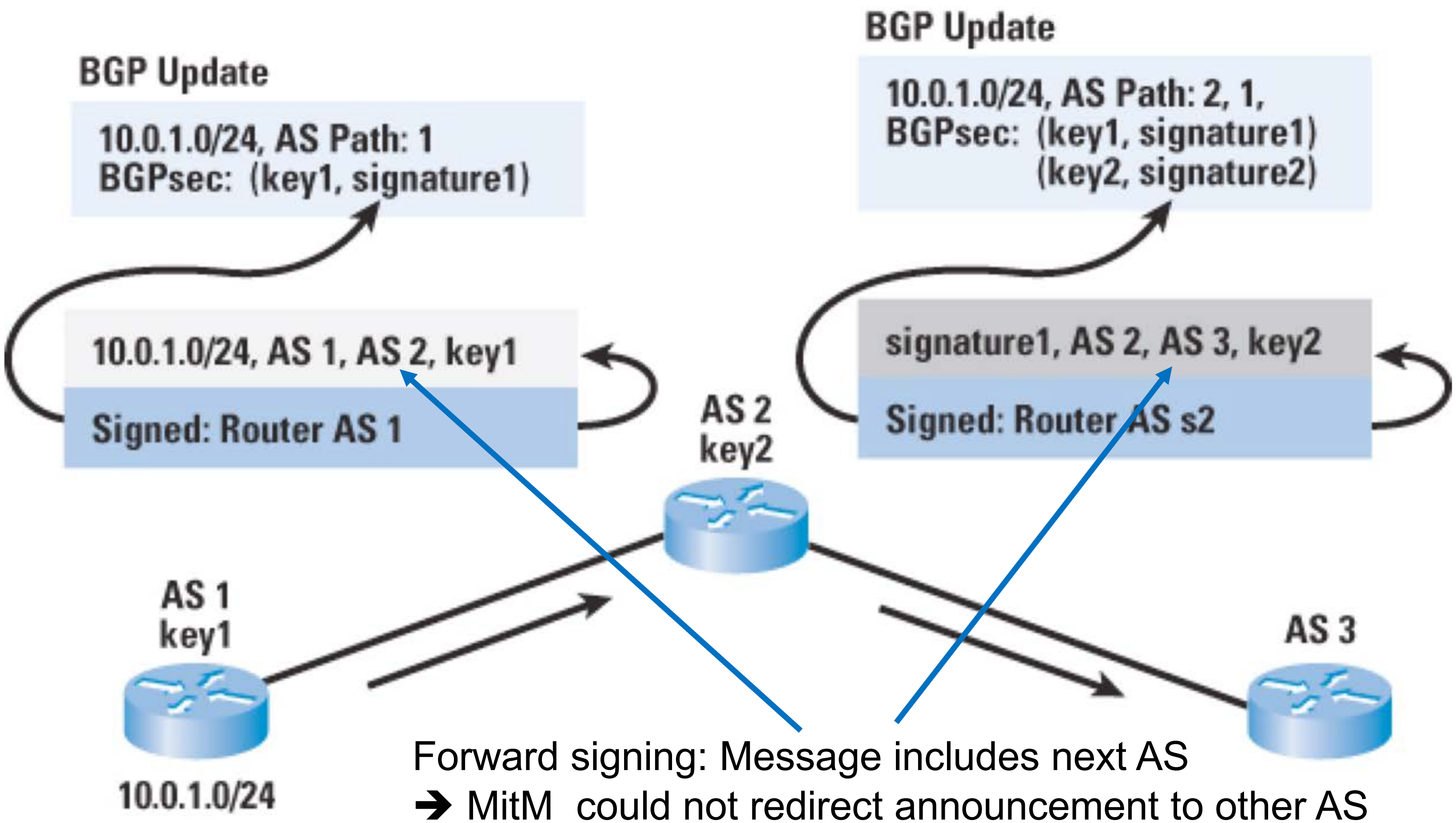
AS Path Validation

- Goal: Validate authenticity of an ***AS Path***
  – Does sequence of ASs in AS Path represent the actual propagation path of the BGP route object?



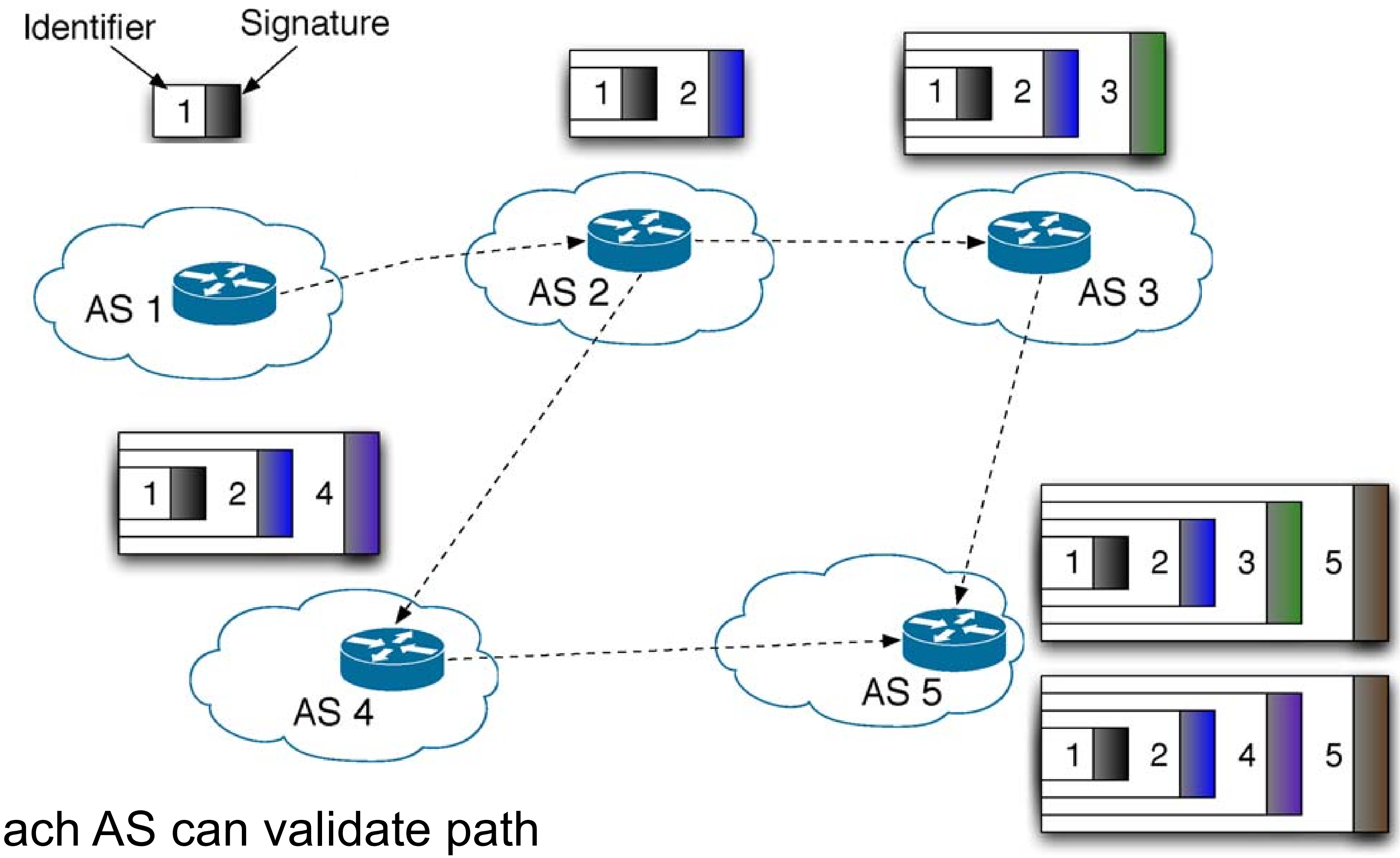
AS Path Validation

- Each router adds signature to BGP update
- Address prefix, own AS number, next AS, own public key



Source: Geoff Huston, Randy Bush, Securing BGP, The Internet Protocol Journal, Volume 14, No. 2, 2011

Nested Signatures



➔ Each AS can validate path

Signed BGP update contains *next AS* to whom update is sent ➔ nested signature cannot be removed without receiver noticing inconsistence

Source: Butler et al., "A Survey of BGP Security Issues and Solutions," *Proceedings of the IEEE*, vol.98, no.1, 2010

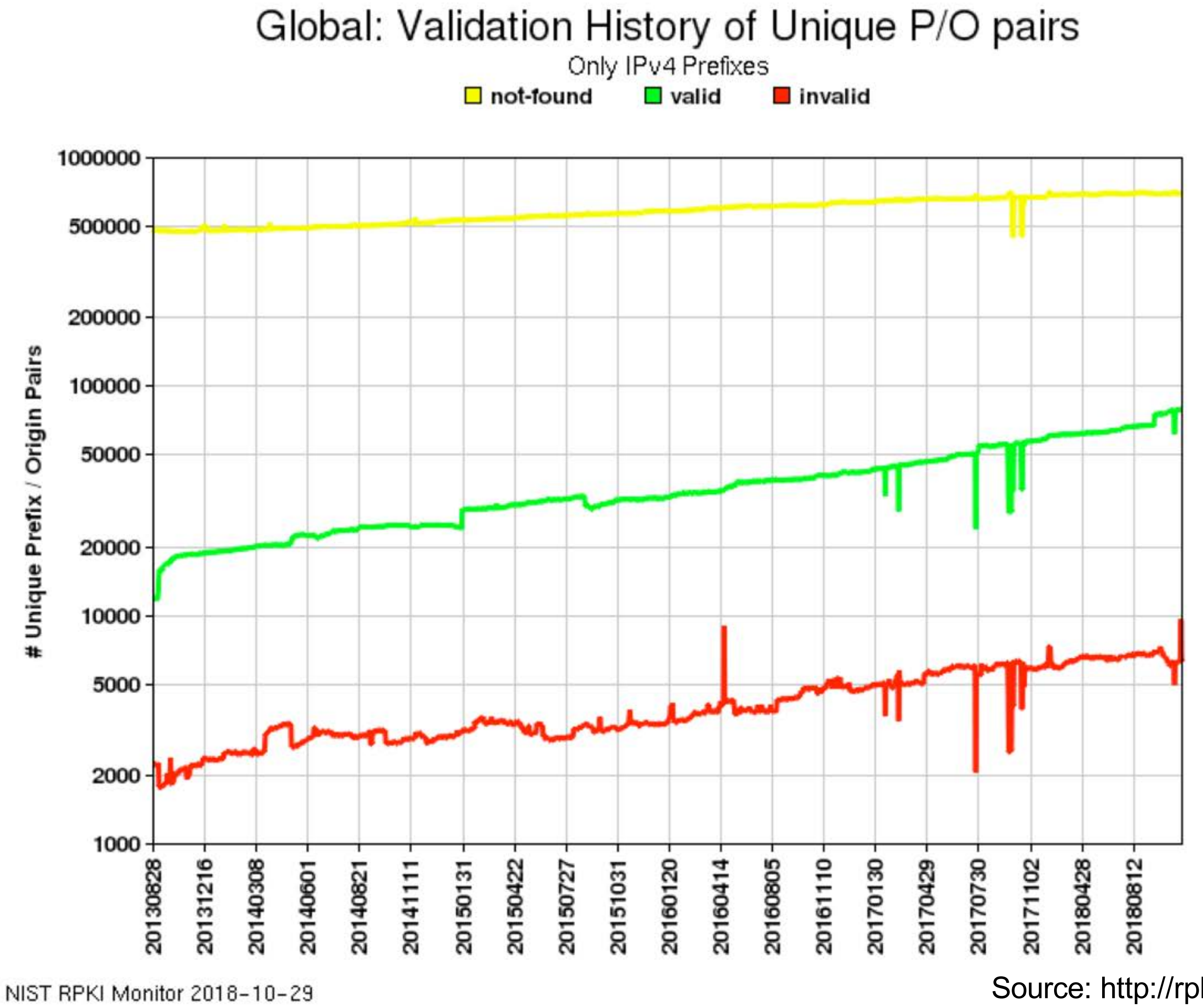
Problems with BGPSEC

- Ensuring the accuracy of registries
  - Information about address ownership, delegation
- Resource consumption
  - Computational requirements for PKI operations
  - Encryption, key exchange, validation of certificates
  - High amount of messages
  - High amount of potential signers
  - ➔ high costs for upgrading routers
- Additional complexity
  - Administrators need expertise

BGP Security Today

- (partially) deployed solutions
  - MD5 MAC
  - Filtering, routing registries
  - Generalized TTL Security Mechanism (GTSM)
- Problem: huge complex system
  - > 54.000 ASes, many interconnections
  - Core of the Internet
- Now: BGPSEC Deployment pushed
- ROA deployment started
  - Regional Registries offer support, trainings
- Path Validation
  - Computational expensive ➔ deployment difficult

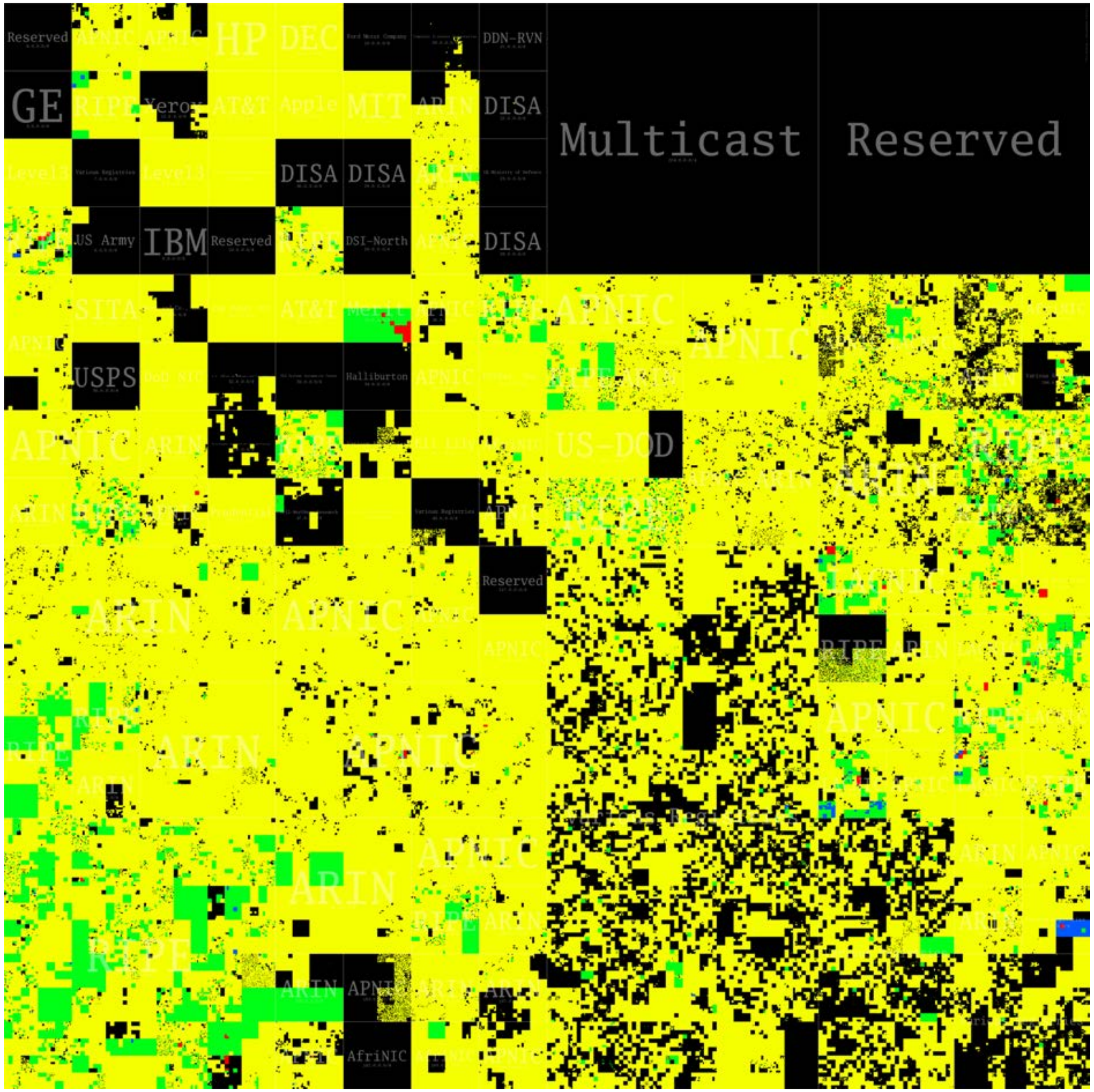
Route Origin Authorization (ROA) Deployment



Source: http://rpki-monitor.antd.nist.gov/

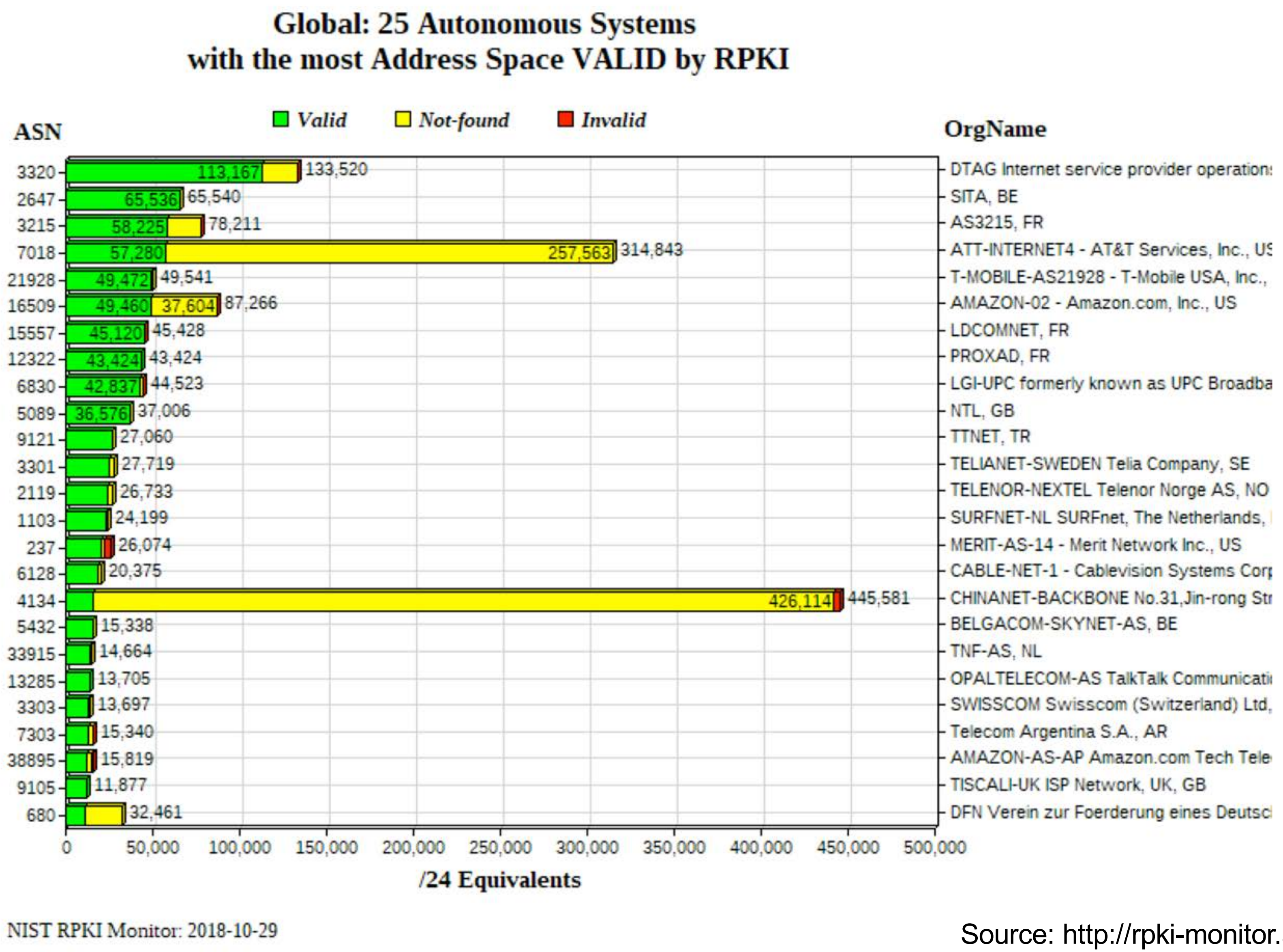
ROA Deployment (IPv4 Address Space)

Green - valid ROA
Yellow – ROA not found
Red – invalid ROA
Black – not observed in BGP
Blue – not observed in trace data,
but covered by ROA



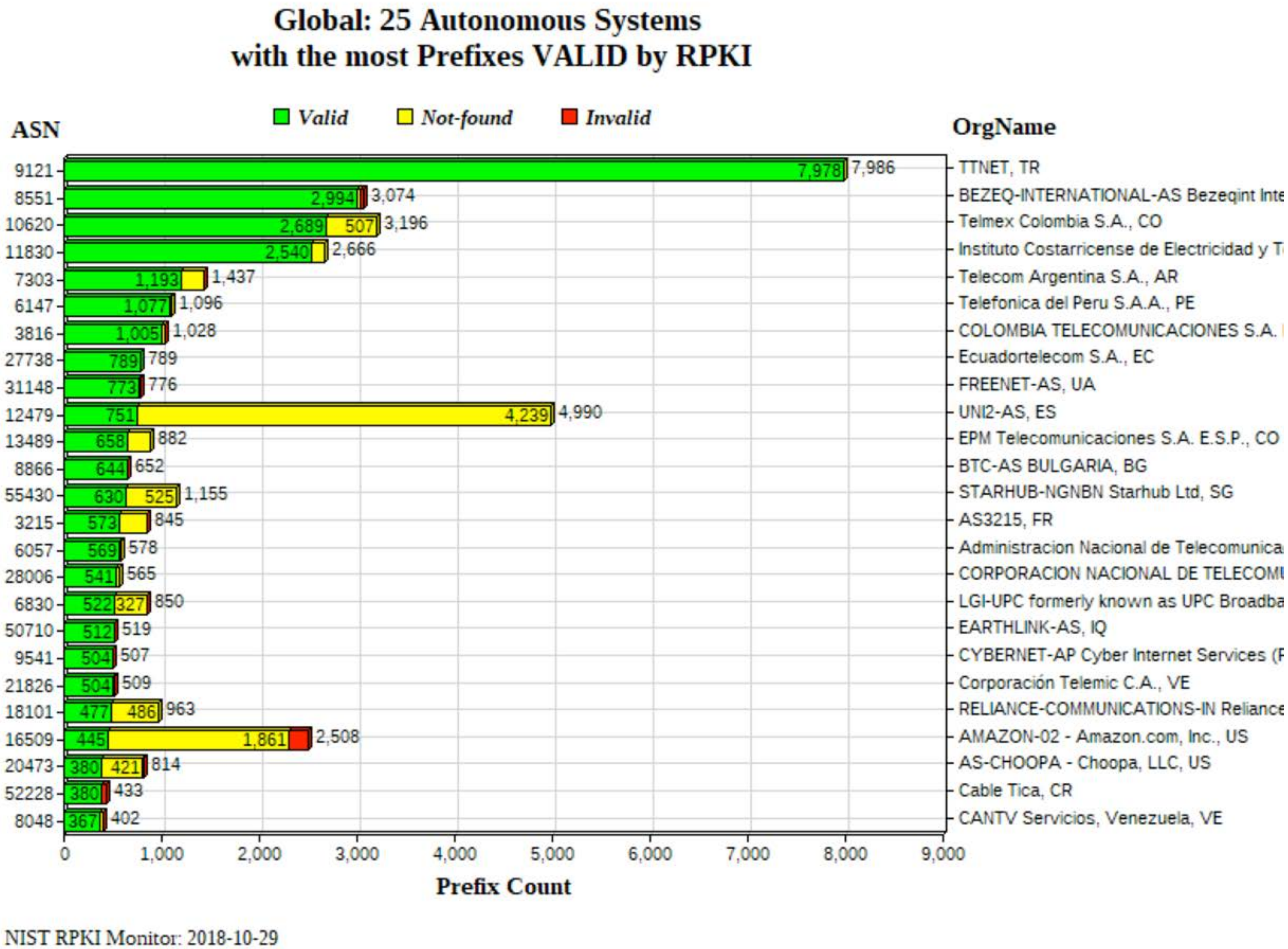
Source: http://rpki-monitor.antd.nist.gov/

ROA Deployment per AS (2018): Address Space



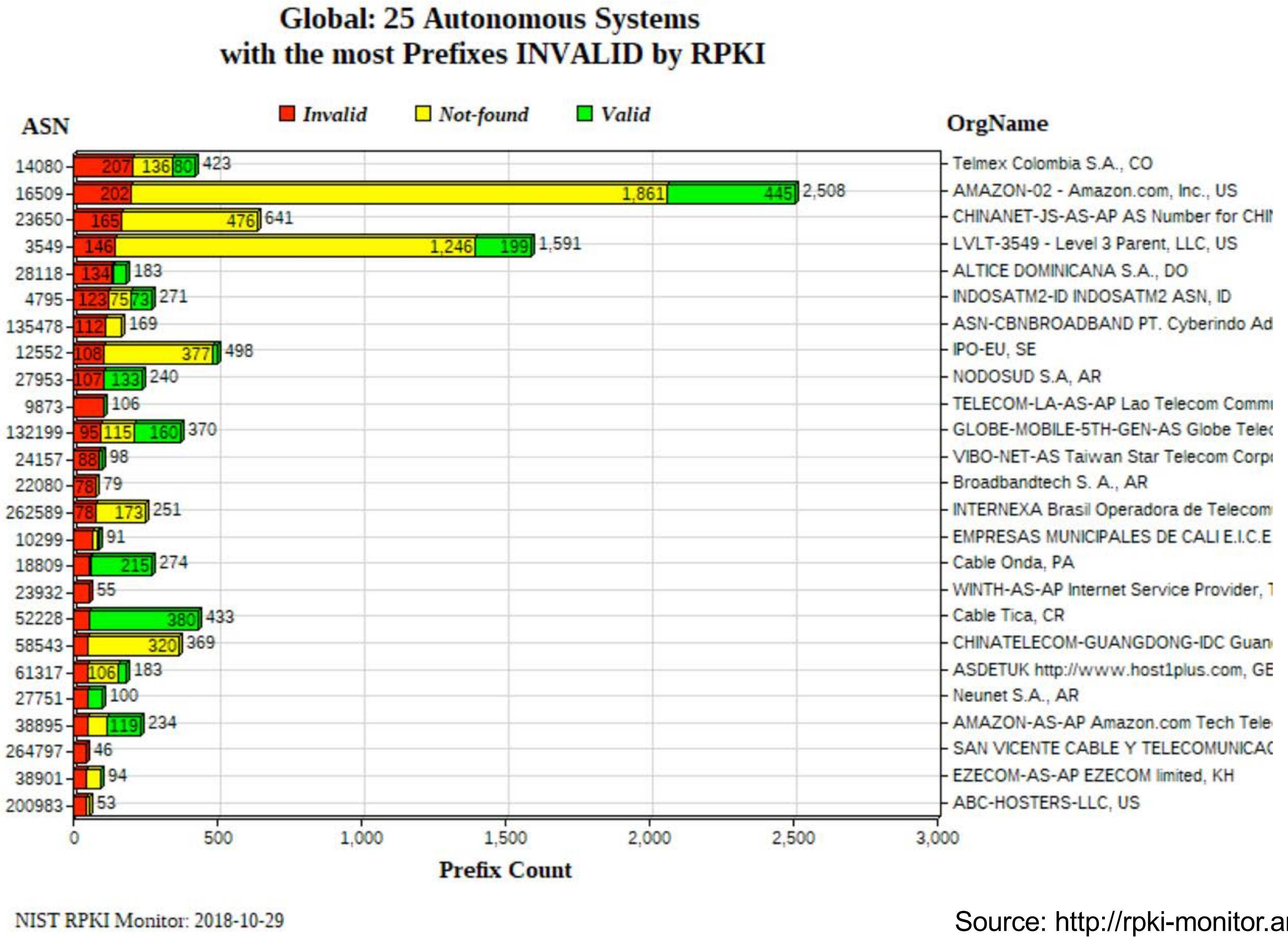
Source: http://rpki-monitor.antd.nist.gov/

ROA Deployment per AS (2018): VALID Prefixes



Source: http://rpki-monitor.antd.nist.gov/

ROA Deployment per AS (2018): INVALID Prefixes



Source: http://rpki-monitor.antd.nist.gov/

BGPSEC Summary

- Resource Public Key Infrastructure (RPKI)
  - Originates and distributes certificates to bind address range to AS number
  - Uses address allocation infrastructure
- Route Origin Authorization (ROA)
  - Authorizes an AS to announce address range
  - Address owner can issue ROAs for other ASes
- AS Path Validation
  - Validate that sequence of ASs in AS Path element is the actual propagation path of BGP message
  - Using nested signatures
- Still work in progress

Thank you!