BGP Security

Amir Herzberg

University of Connecticut

November 18, 2024

Internet Routing is a large challenge

- The Internet is composed of over *80,000* independently-managed Autonomous Systems (ASes), mostly for-profit.
- IPv4 and IPv6 addresses are split to prefixes, each owned by given AS.
  - Over *1M* IPv4 and 200K IPv6 prefix/origin pairs announced.
  - Topology, ownership of prefixes and announcements change.
- Routing coordinated between ASes, each with their own goals.
  - Provide good service to customers, maximize revenues, minimize costs
  - A free market economy: no centralized planning, controls.

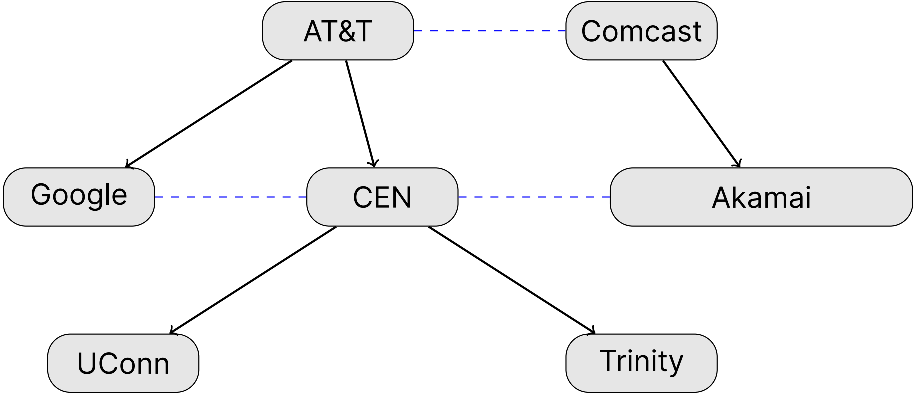
[1] RFC-4271

Internet Routing is a large challenge

- The Internet is composed of over *80,000* independently-managed Autonomous Systems (ASes), mostly for-profit.
- IPv4 and IPv6 addresses are split to prefixes, each owned by given AS.
  - Over *1M* IPv4 and 200K IPv6 prefix/origin pairs announced.
  - Topology, ownership of prefixes and announcements change.
- Routing coordinated between ASes, each with their own goals.
  - Provide good service to customers, maximize revenues, minimize costs
  - A free market economy: no centralized planning, controls.
- IETF solution: separate Inter-AS Routing and Intra-AS Routing protocols
  - Each AS can use its own Intra-AS Routing
  - Inter-AS Routing done by the **Border Gateway Protocol (BGP)**[1].

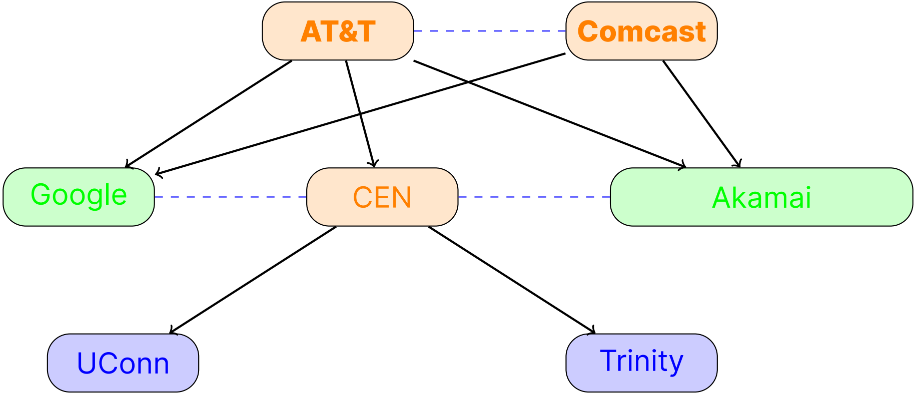
[1] RFC-4271

BGP: customer-provider and bilateral-peers AS relationships



Types of ASes

Transit: have customer (often also peers, providers)
Stubs: have only one provider, no customers, rarely peers
Multi-home: no customers; at least two providers, possibly peers
Top-tier: transit ASes with no providers, usually all peered



Forwarding Packets and FIB

- Routers (aka gateways) **forward** IP packets toward destination network
- Networks identified by **prefix**: $1.2/16 = \{1.2.x.y\}_{x,y=0}^{255}$
- **Forwarding Info-Base (FIB)**: table mapping prefixes to next-AS (or router)
    1. Can't have two mappings for the same prefix
    2. But can have entry for prefix (1.2/16) and subprefix (1.2.3/24)
    3. Routers use the **most specific** prefix for dest-IP

'

Forwarding Packets and FIB

- Routers (aka gateways) **forward** IP packets toward destination network
- Networks identified by **prefix**: $1.2/16 = \{1.2.x.y\}_{x,y=0}^{255}$
- **Forwarding Info-Base (FIB)**: table mapping prefixes to next-AS (or router)
  1. Can't have two mappings for the same prefix
  2. But can have entry for prefix (1.2/16) and subprefix (1.2.3/24)
  3. Routers use the **most specific** prefix for dest-IP

‘

Prefix	Next-AS
1.2/16	1
1.2.3/24	2
5.6.8/22	3

Forwarding Packets and FIB

- Routers (aka gateways) **forward** IP packets toward destination network
- Networks identified by **prefix**: $1.2/16 = \{1.2.x.y\}_{x,y=0}^{255}$
- **Forwarding Info-Base (FIB)**: table mapping prefixes to next-AS (or router)
  1. Can't have two mappings for the same prefix
  2. But can have entry for prefix (1.2/16) and subprefix (1.2.3/24)
  3. Routers use the **most specific** prefix for dest-IP

'

Prefix	Next-AS
1.2/16	1
1.2.3/24	2
5.6.8/22	3

Destination 1.2.3.4, route to AS ____

Forwarding Packets and FIB

- Routers (aka gateways) **forward** IP packets toward destination network
- Networks identified by **prefix**: $1.2/16 = \{1.2.x.y\}_{x,y=0}^{255}$
- **Forwarding Info-Base (FIB)**: table mapping prefixes to next-AS (or router)
  1. Can't have two mappings for the same prefix
  2. But can have entry for prefix (1.2/16) and subprefix (1.2.3/24)
  3. Routers use the **most specific** prefix for dest-IP

'

Prefix	Next-AS
1.2/16	1
1.2.3/24	2
5.6.8/22	3

Destination 1.2.3.4, route to AS ____
Designation 1.2.5.6, route to AS ____

Forwarding Packets and FIB

- Routers (aka gateways) **forward** IP packets toward destination network
- Networks identified by **prefix**: $1.2/16 = \{1.2.x.y\}_{x,y=0}^{255}$
- **Forwarding Info-Base (FIB)**: table mapping prefixes to next-AS (or router)
    1. Can't have two mappings for the same prefix
    2. But can have entry for prefix (1.2/16) and subprefix (1.2.3/24)
    3. Routers use the **most specific** prefix for dest-IP

'

Prefix	Next-AS
1.2/16	1
1.2.3/24	2
5.6.8/22	3

Destination 1.2.3.4, route to AS ____
Designation 1.2.5.6, route to AS ____
Destination 5.6.9.1, route to AS ____

Forwarding Packets and FIB

- Routers (aka gateways) **forward** IP packets toward destination network
- Networks identified by **prefix**: $1.2/16 = \{1.2.x.y\}_{x,y=0}^{255}$
- **Forwarding Info-Base (FIB)**: table mapping prefixes to next-AS (or router)
  1. Can’t have two mappings for the same prefix
  2. But can have entry for prefix (1.2/16) and subprefix (1.2.3/24)
  3. Routers use the **most specific** prefix for dest-IP

‘

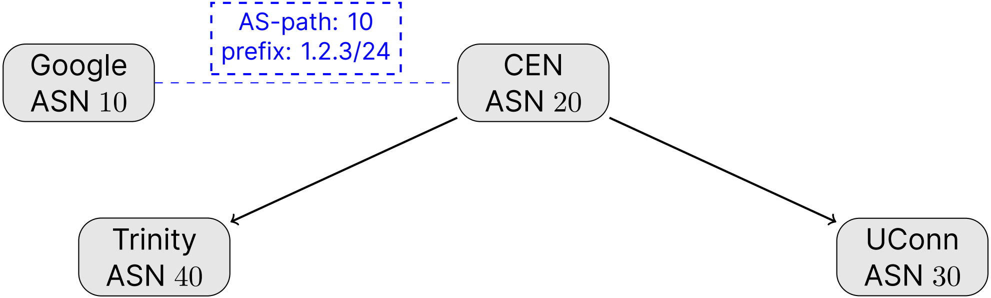
Prefix	Next-AS
1.2/16	1
1.2.3/24	2
5.6.8/22	3

Destination 1.2.3.4, route to AS ____
Designation 1.2.5.6, route to AS ____
Destination 5.6.9.1, route to AS ____
Destination 7.8.9.2,____________________

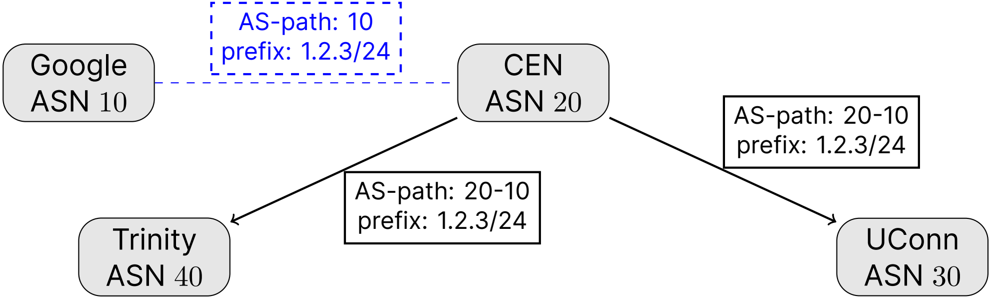
Routing: Creating the Forwarding Tables (FIBs)

- Routers (gateways) receive route-announcements from neighboring routers, put (best?) route to each prefix in FIB
- Routing protocol receives, processes and sends announcements
- Intra-Domain Routing: routing within the AS
- Inter-Domain Routing: routing to/from other ASes
- BGP (Border Gateway Protocol): the Internet's standard Inter-Domain Routing protocol

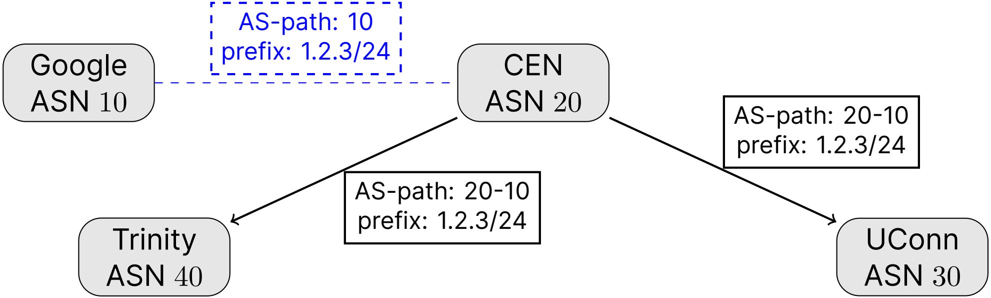
BGP Announcements



BGP Announcements



BGP Announcements



- Ignore incoming announcement if it contains your ASN (loop prevention)
- Policy determines which incoming announcement to use and which (if any) to **export** (send) to each neighbor
- Policy determined by AS, but expected to be **sensible**
- These considerations imply most policy choices!

BGP policy: economics, performance, connectivity

- ASes pay their providers based on amount of traffic (send and received!)
- No payments for traffic between bilateral peers
- Shorter path (less ASes) is often also faster (less routers, delay)
- Most important: **connectivity** (to your AS and your customers)

Valley-Free (Gao-Rexford) BGP Policies

- Prefer announcements based on relationships:

  Best: announcements from customers (get paid!)
  Ok: announcements from bilateral peers
  Least: announcements from providers (pay)

- Among these, prefer announcements with shortest AS-path

Valley-Free (Gao-Rexford) BGP Policies

- Prefer announcements based on relationships:

  Best: announcements from customers (get paid!)

  Ok: announcements from bilateral peers

  Least: announcements from providers (pay)

- Among these, prefer announcements with shortest AS-path
- Continue using current announcement if as good as new one
- Tie-break: if all the same, use announcement from AS with lower ASN

Valley-Free (Gao-Rexford) BGP Policies

- Prefer announcements based on relationships:

  Best: announcements from customers (get paid!)
  Ok: announcements from bilateral peers
  Least: announcements from providers (pay)

- Among these, prefer announcements with shortest AS-path
- Continue using current announcement if as good as new one
- Tie-break: if all the same, use announcement from AS with lower ASN
- **Export** the chosen announcement to all customers; if it is from a customer, export to peers and to one or all providers.
  - By default (and in most works): export to all providers

BGP Traffic Engineering (TE)

- Methods for prefix-owner to influence selection of path:
- **Prepending:** announce as usual to preferred provider (e.g., AS-path:10) and by prepending to depreferred provider (AS-path 10-10-10)
- **Announce only to preferred provider**; announce to other provider only if/when needed
- **Communities:** optional fields for customer AS to signal requests to provider AS (e.g., do not announce to AS X)

Secure BGP (Inter-Domain Routing) is a really large challenge

- BGP coordinates routing between many ASes
  - A free market economy: no centralized planning, controls.
  - BGP allows each AS to choose its own policy.
  - Often conflicting goals; many attacks, failures!
- Some basic rules should be respected:

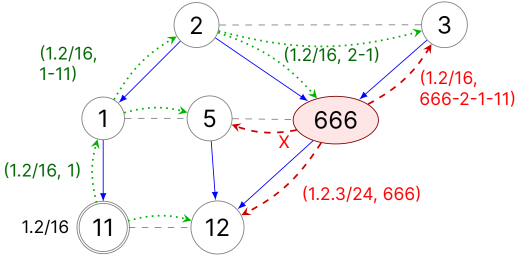
Secure BGP (Inter-Domain Routing) is a really large challenge

- BGP coordinates routing between many ASes
  - A free market economy: no centralized planning, controls.
  - BGP allows each AS to choose its own policy.
  - Often conflicting goals; many attacks, failures!
- Some basic rules should be respected:
  - Only announce your prefixes and relayed announcements
  - Preserve the integrity of relayed announcements
  - Valley-free routing: maximize profits and customer-connectivity
- BGP attacks are forbidden behaviors.

BGP Mis-Routing Attacks

BGP lacks authentication. BGP sessions are often authenticated against MitM (using TLS, IPSec,...) but BGP is still vulnerable to rogue AS attacks:

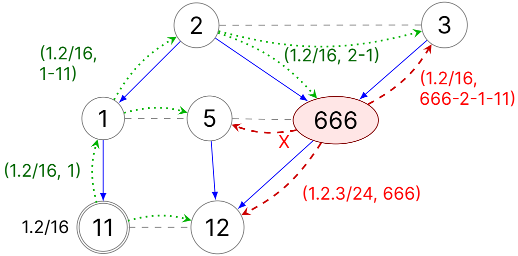
- Route Leak (valley): up to AS 3
- Prefix Hijack: X=(1.2/16, 666) to AS 5
- Subprefix Hijack: (1.2.3/24,666) to AS 12
- Origin Hijack: X=(1.2/16, 666-11) to AS 5
- Path Manipulation: X=(1.2/16, 666-2-11)



BGP Mis-Routing Attacks

BGP lacks authentication. BGP sessions are often authenticated against MitM (using TLS, IPSec,...) but BGP is still vulnerable to rogue AS attacks:

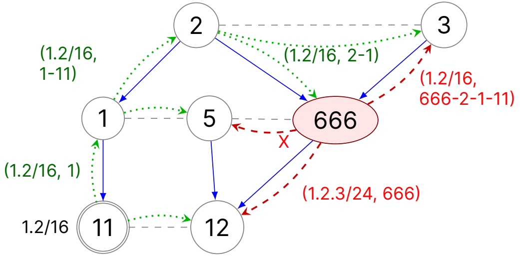
- Route Leak (valley): up to AS 3
- Prefix Hijack: X=(1.2/16, 666) to AS 5
- Subprefix Hijack: (1.2.3/24,666) to AS 12
- Origin Hijack: X=(1.2/16, 666-11) to AS 5
- Path Manipulation: X=(1.2/16, 666-2-11)
  - Attribute Manipulation: X=(1.2/16, 666-2-1-11, blackhole) to AS 5



BGP Mis-Routing Attacks

BGP lacks authentication. BGP sessions are often authenticated against MitM (using TLS, IPSec,...) but BGP is still vulnerable to rogue AS attacks:

- Route Leak (valley): up to AS 3
- Prefix Hijack: X=(1.2/16, 666) to AS 5
- Subprefix Hijack: (1.2.3/24,666) to AS 12
- Origin Hijack: X=(1.2/16, 666-11) to AS 5
- Path Manipulation: X=(1.2/16, 666-2-11)



- Attribute Manipulation: X=(1.2/16, 666-2-1-11, blackhole) to AS 5
- Attack or misconfiguration (‘fat fingers’)?
  - Motivations for attacks: MitM, eavesdrop, DoS, spam/phishing, deanonymization, DNS poison, ...

A Brief, Partial History of BGP Security (not to scale)

1989

RFC 1105 A Border Gateway Protocol (BGP)
Security Problems in the TCP/IP Protocol Suite

1994

RFC 1654 A Border Gateway Protocol 4 (BGP-4)

1999

Secure Border Gateway Protocol (S-BGP)

2001

Stable Internet Routing without Global Coordination

2003

Origin Authentication in Interdomain Routing
Securing BGP through Secure Origin BGP (soBGP)

2004

Evaluation of Efficient Security for BGP Route Announcements using Parallel Simulation
SPV: Secure Path Vector Routing for Securing BGP
Listen and Whisper: Security Mechanisms for BGP

2005

Aggregated Path Authentication for Efficient BGP Security

2006

RFC 4272 BGP Security Vulnerabilities Analysis
PHAS: a Prefix Hijack Alert System

2007

On Interdomain Routing Security and Pretty Secure BGP (psBGP)

2008

Autonomous Security for Autonomous Systems

2009

Netreview: Detecting When Interdomain Routing Goes Wrong

A Brief, Partial History of BGP Security (not to scale)

2010

- *A Survey of BGP Security Issues and Solutions*
- *How Secure are Secure Interdomain Routing Protocols?*

2011

- *Let the Market Drive Deployment: A Strategy for Transitioning to BGP Security*
- *Having your Cake and Eating it too: Routing Security with Privacy Protections*
- *Preventing Attacks on BGP Policies: One Bit is Enough*

2012

- *RFC 6480 An Infrastructure to Support Secure Internet Routing*
- *RFC 6481 A Profile for Resource Certificate Repository Structure*
- *Private and Verifiable Interdomain Routing Decisions*
- *A new approach to Interdomain Routing based on Secure Multi-party Computation*

2013

- *RFC 6811 BGP Prefix Origin Validation*
- *BGP Security in Partial Deployment: Is the Juice worth the Squeeze?*
- *On the Risk of Misbehaving RPKI Authorities*
- *A Survey of Interdomain Routing Policies*

2014

- *Why is it Taking so Long to Secure Internet Routing?*
- *RFC 7132 Threat Model for BGP Path Security*
- *PEERING: an AS for us*
- *A Survey of Interdomain Routing Policies*

2015

- *Secure Routing for Future Communication Networks*
- *Investigating Interdomain Routing Policies in the Wild*
- *Self-reliant Detection of Route Leaks in Inter-domain Routing*

A Brief, Partial History of BGP Security (not to scale)

2016

- RFC 7908 Problem Definition and Classification of BGP Route Leaks
- *Jumpstarting BGP Security with Path-End Validation*
- *Rethinking Security for Internet Routing*
- *NTT Peer Locking*

2017

- RFC 8205 BGPsec Protocol Specification
- *Are We There Yet? On RPKI's Deployment and Security*
- *Design and Analysis of Optimization Algorithms to Minimize Cryptographic Processing in BGP Security Protocols*
- *The SCION Internet Architecture*

2018

- *RFC 8374 BGPsec Design Choices and Summary of Supporting Discussions*
- *Practical Experience: Methodologies for Measuring Route Origin Validation*
- *Towards a Rigorous Methodology for Measuring Adoption of RPKI Route Validation and Filtering*
- *University of Oregon Route Views Project*
- *The State of Affairs in BGP Security: A Survey of Attacks and Defenses*

2019

- *Resilient Interdomain Traffic Exchange: BGP Security and DDoS Mitigation*
- *RPKI is Coming of Age: A Longitudinal Study of RPKI Deployment and Invalid Route Origins*
- *SICO: Surgical Interception Attacks by Manipulating BGP Communities*

2020

- *Limiting the Power of RPKI Authorities*
- *DISCO: Sidestepping RPKI's Deployment Barriers*
- *On Measuring RPKI Relying Parties*
- *Peerlock: Flexsealing BGP*

2021

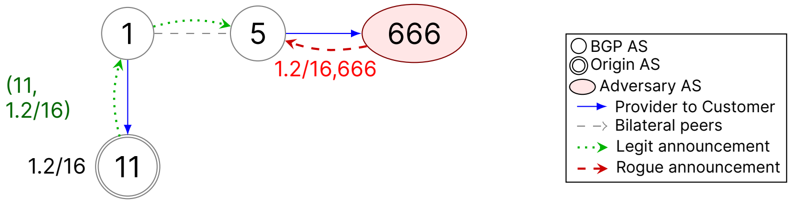
- *Revisiting RPKI Route Origin Validation on the Data Plane*
- *ROV++: Improved Deployable Defense Against BGP Hijacking*
- *The Hijackers Guide to the Galaxy:Off-Path Taking Over Internet Resources*

2024

- *BGP-iSec*
- *ASPA*

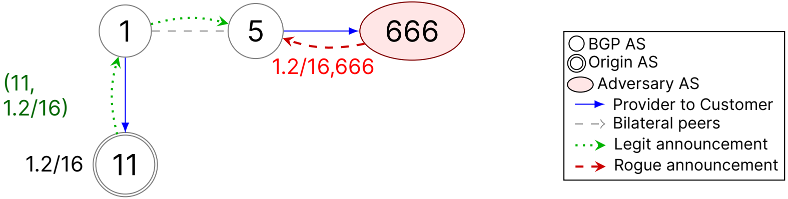
BGP Security and Attacks

BGP sessions are often authenticated against MitM (e.g., IPsec)
But Rogue AS is often able to intercept traffic
Example: prefix hijack intercepts traffic sent from AS 5 to 1.2/16



BGP Security and Attacks

BGP sessions are often authenticated against MitM (e.g., IPsec)
But Rogue AS is often able to intercept traffic
Example: prefix hijack intercepts traffic sent from AS 5 to 1.2/16



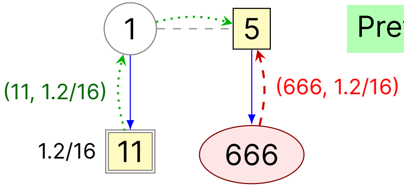
What defenses can foil this attack?

Route Origin Authorization (ROA)

- Prefix-owners sign Route Origin Authorization (ROA), defining a valid origin-AS for each prefix
- Assume ROA for 1.2/16, origin AS 11. Following announcement are invalid:
    - Announcements with origin AS 666 and prefix 1.2/16: wrong origin AS,
    - and with origin AS 11, and prefix ??

Route Origin Authorization (ROA)

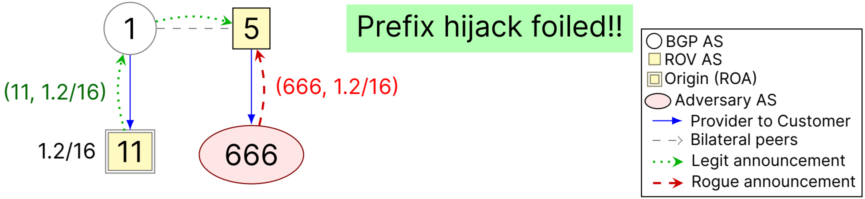
- Prefix-owners sign Route Origin Authorization (ROA), defining a valid origin-AS for each prefix
- Assume ROA for 1.2/16, origin AS 11. Following announcement are invalid:
  - Announcements with origin AS 666 and prefix 1.2/16: wrong origin AS,
  - and with origin AS 11, and prefix 1.2.3/24: no ROA for subprefix



Prefix hijack foiled!!

Route Origin Authorization (ROA)

- Prefix-owners sign Route Origin Authorization (ROA), defining a valid origin-AS for each prefix
- Assume ROA for 1.2/16, origin AS 11. Following announcement are invalid:
    - Announcements with origin AS 666 and prefix 1.2/16: wrong origin AS,
    - and with origin AS 11, and prefix 1.2.3/24: no ROA for subprefix
    - Prefix owner can sign multiple ROAs for the same prefix (different ASes): all allowed as origin.



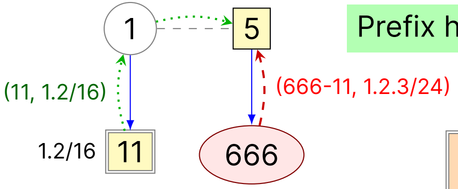
Prefix hijack foiled!!

Route Origin Authorization (ROA)

- Prefix-owners sign Route Origin Authorization (ROA), defining a valid origin-AS for each prefix
- Assume ROA for 1.2/16, origin AS 11. Following announcement are invalid:
  - Announcements with origin AS 666 and prefix 1.2/16: wrong origin AS,
  - and with origin AS 11, and prefix 1.2.3/24: no ROA for subprefix
  - Prefix owner can sign multiple ROAs for the same prefix (different ASes): all allowed as origin.
  - ROA has optional parameter max-length$= l$; in this case, subprefixes with length up to l are valid
  - E.g., with a ROA for 1.2/16 with max-length$= 22$ and origin AS 11, announcement (11, 1.2.8/22) is valid (but 1.2.3/24 is invalid)
  - Can reduce number of ROAs but vulnerable if not all prefixes allowed, therefore, avoid it [RFC9319]

Route Origin Authorization (ROA)

- Prefix-owners sign Route Origin Authorization (ROA), defining a valid origin-AS for each prefix
- Assume ROA for 1.2/16, origin AS 11. Following announcement are invalid:
  - Announcements with origin AS 666 and prefix 1.2/16: wrong origin AS,
  - and with origin AS 11, and prefix 1.2.3/24: no ROA for subprefix
- Routers deploying Route Origin Validation (ROV) drop invalid announcements, mitigating prefix hijacks



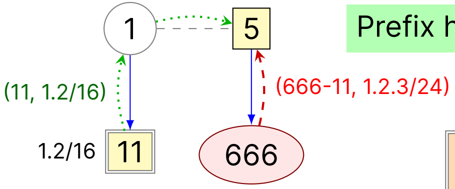
Prefix hijack foiled!!

What about subprefix hijacks?

This example is prevented...
But sometimes, attack succeeds!

Route Origin Validation (ROV)

- Prefix-owners sign Route Origin Authorization (ROA), defining a valid origin-AS for each prefix
- Assume ROA for 1.2/16, origin AS 11. Following announcement are invalid:
  - Announcements with origin AS 666 and prefix 1.2/16: wrong origin AS,
  - and with origin AS 11, and prefix 1.2.3/24: no ROA for subprefix
- Routers deploying Route Origin Validation (ROV) drop invalid announcements, mitigating prefix hijacks

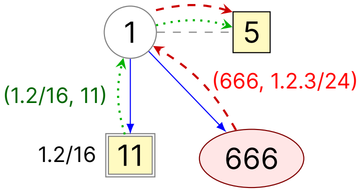


Prefix hijack foiled!!

What about subprefix hijacks?

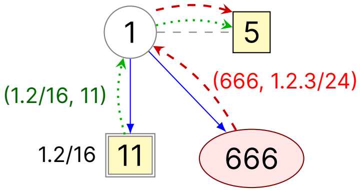
This example is prevented...
But sometimes, attack succeeds!

Partially-adopted Route Origin Validation (ROV) may fail against subprefix hijacks



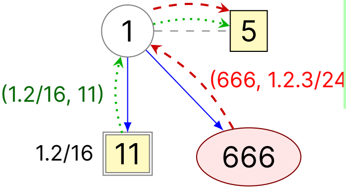
- AS 1 doesn't adopt ROV, hence, forwards (666, 1.2.3/24) to AS 5.
- Even if AS 5 adopts ROV, and drops (666, 1.2.3/24), it would route to AS 1 packets with dest-IP in 1.2/16, including in 1.2.3/24; and AS 1 routes to the attacker packets with dest-IP in 1.2.3/24 (IP always routes to the most specific prefix in the routing table)

Partially-adopted Route Origin Validation (ROV) may fail against subprefix hijacks



- AS 1 doesn't adopt ROV, hence, forwards (666, 1.2.3/24) to AS 5.
- Even if AS 5 adopts ROV, and drops (666, 1.2.3/24), it would route to AS 1 packets with dest-IP in 1.2/16, including in 1.2.3/24; and AS 1 routes to the attacker packets with dest-IP in 1.2.3/24 (IP always routes to the most specific prefix in the routing table)
- Prevent with **ROV++** (ROV+ 'never send towards subprefix hijack')

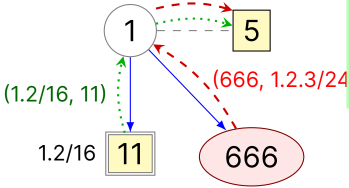
ROV++ [NDSS21] foils this (and most) subprefix hijacks!



Suppose AS 5 adopts ROV++.
It would blackhole traffic to 1.2.3/2
rather than send via AS 1
⇒ **subprefix hijack foiled!!**

- AS 1 doesn't adopt ROV, hence, forwards (666, 1.2.3/24) to AS 5.
- Even if AS 5 adopts ROV, and drops (666, 1.2.3/24), it would route to AS 1 packets with dest-IP in 1.2/16, including in 1.2.3/24; and AS 1 routes to the attacker packets with dest-IP in 1.2.3/24
- Prevent with **ROV++** (ROV+ 'never send towards subprefix hijack')

ROV++ [NDSS21] foils this (and most) subprefix hijacks!

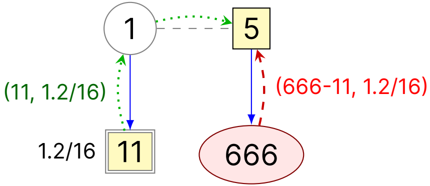


Suppose AS 5 adopts ROV++.
It would blackhole traffic to 1.2.3/2
rather than send via AS 1
⇒ **subprefix hijack foiled!!**

- AS 1 doesn't adopt ROV, hence, forwards (666, 1.2.3/24) to AS 5.
- Even if AS 5 adopts ROV, and drops (666, 1.2.3/24), it would route to AS 1 packets with dest-IP in 1.2/16, including in 1.2.3/24; and AS 1 routes to the attacker packets with dest-IP in 1.2.3/24
- Prevent with **ROV++** (ROV+ 'never send towards subprefix hijack')
  - Attackers expected to switch to **post-ROV** attacks

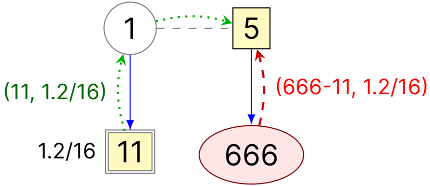
Route Origin Validation (ROV) may fail against Origin Hijacks

- Origin hijack: attacker exports announcement with AS-path containing itself and the legitimate origin, e.g., (666-11, 1.2/16), i.e., as if it received it from the origin
- ROV (and ROV++) evaluate (666-11, 1.2/16) as valid
- The AS-path contains one more AS (cf. prefix hijack) ⇒ less likely to 'win'
  - BGP ASes prefer an announcement from customer, then peer, then provider; if there are multiple announcements from same 'type' (e.g. customer), prefer shorter AS-path.



Route Origin Validation (ROV) may fail against Origin Hijacks

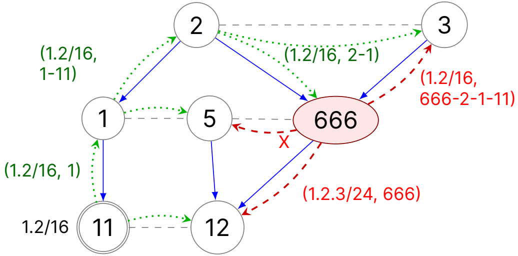
- Origin hijack: attacker exports (666-11, 1.2/16)
- ROV (and ROV++) evaluate (666-11, 1.2/16) as valid
- AS 5 receives (1-11, 1.2/16) from peer (AS 1) and (666-11, 1.2/16) from customer (AS 666). Customer routes are preferred ⇒ traffic to 1.2/16 sent to AS 666!
- The AS-path contains one more AS (cf. prefix hijack) ⇒ less likely to 'win'
- If ROV/ROV++ is only partially adopted, attacker may combine this with a subprefix hijack



BGP Mis-Routing Attacks

BGP lacks authentication. BGP sessions are often authenticated against MitM (using TLS, IPSec,...) but BGP is still vulnerable to rogue AS attacks:

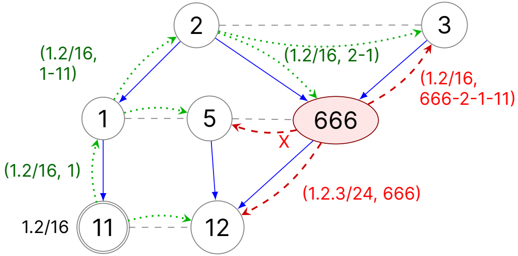
- Prefix Hijack: X=(1.2/16, 666) to AS 5
- Subprefix Hijack: (1.2.3/24,666) to AS 12
- Route Leak (valley): up to AS 3
- Origin Hijack: X=(1.2/16, 666-11) to AS 5
- Path Manipulation: X=(1.2/16, 666-2-11)



BGP Mis-Routing Attacks

BGP lacks authentication. BGP sessions are often authenticated against MitM (using TLS, IPSec,...) but BGP is still vulnerable to rogue AS attacks:

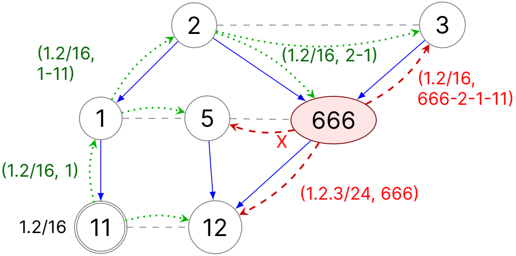
- Prefix Hijack: X=(1.2/16, 666) to AS 5
- Subprefix Hijack: (1.2.3/24,666) to AS 12
- Route Leak (valley): up to AS 3
- Origin Hijack: X=(1.2/16, 666-11) to AS 5
- Path Manipulation: X=(1.2/16, 666-2-11)
  - Attribute Manipulation: X=(1.2/16, 666-2-1-11, blackhole) to AS 5



BGP Mis-Routing Attacks

BGP lacks authentication. BGP sessions are often authenticated against MitM (using TLS, IPSec,...) but BGP is still vulnerable to rogue AS attacks:

- Prefix Hijack: X=(1.2/16, 666) to AS 5
- Subprefix Hijack: (1.2.3/24,666) to AS 12
- Route Leak (valley): up to AS 3
- Origin Hijack: X=(1.2/16, 666-11) to AS 5
- Path Manipulation: X=(1.2/16, 666-2-11)



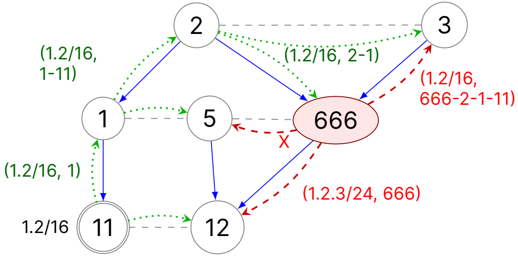
- Attribute Manipulation: X=(1.2/16, 666-2-1-11, blackhole) to AS 5
- Attack or misconfiguration ('fat fingers')?
  - Motivations for attacks: MitM, eavesdrop, DoS, spam/phishing, deanonymization, DNS poison, ...

UCONN
SCHOOL OF ENGINEERING

Post-ROV BGP Mis-Routing Attacks

With complete adoption of ROAs and ROV, prefix and subprefix attacks are eliminated. Remaining threats:

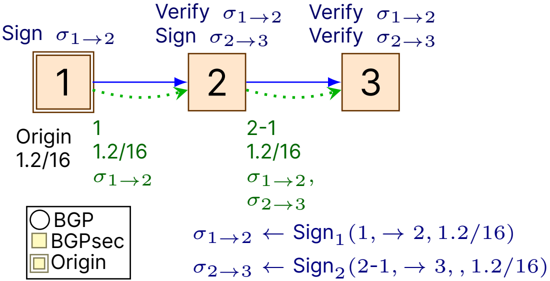
- ~~Prefix Hijack: X=(1.2/16, 666) to AS 5~~
- ~~Subprefix Hijack: (1.2.3/24,666) to AS 12~~
- Route Leak (valley): up to AS 3
- Origin Hijack: X=(1.2/16, 666-11) to AS 5
- Path Manipulation: X=(1.2/16, 666-2-11)



- Attribute Manipulation: X=(1.2/16, 666-2-1-11, blackhole) to AS 5

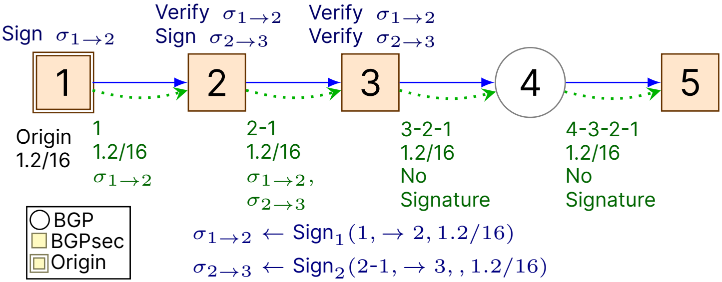
BGPsec (RFC8205): IETF standard against path manipulations.

- ASes sign announcements they export, and validate sigs on incoming announcements
- Add 'next AS' to announcement, e.g., $(2-1, \rightarrow 3, 1.2/16)$
- RPKI contains certificates with ASN and public key of that ASN



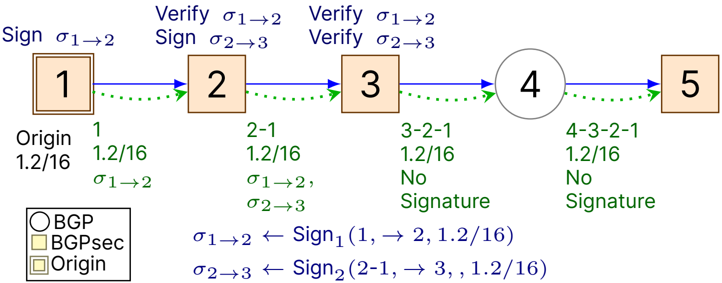
- Attacker can't make a BGPsec-valid origin hijack or other path-manipulations

BGPsec (RFC8205): IETF standard against path manipulations.



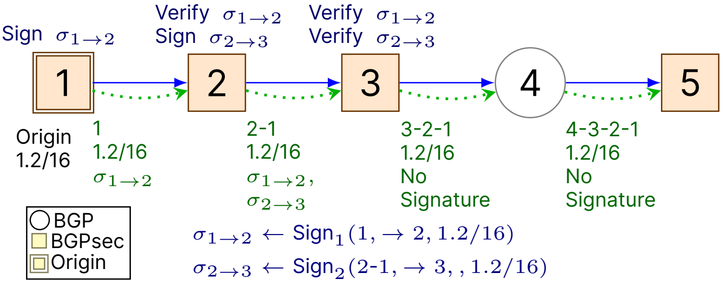
- Attacker can't make a BGPsec-valid origin hijack or other path-manipulations
- BGPsec ASes downgrade to BGP for BGP neighbors
  - E.g, AS 5 will not receive signature, can't validate.
- $\Rightarrow$ Very limited benefits for partial deployment [LychevGS13]

BGPsec (RFC8205): IETF standard against path manipulations.



- BGPsec ASes downgrade to BGP for BGP neighbors
- **Why does BGPsec downgrade to BGP?**

BGPsec (RFC8205): IETF standard against path manipulations.

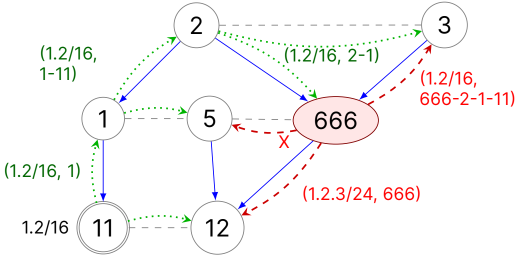


- BGPsec ASes downgrade to BGP for BGP neighbors
- **Why does BGPsec downgrade to BGP?**
- BGPsec ASes do not relay BGPsec info to BGP-only routers.
- Even if they did, rogue AS can omit BGPsec info
    - BGPsec has no registry of adopting ASes
    - And adopting ASes may stop signing at any time

Mis-Routing Attacks in spite of BGPsec (and ROV)

All post-ROV vulnerabilities remain even with global adoption of BGPsec!

- ~~Prefix Hijack: X=(1.2/16, 666) to AS 5~~
- ~~Subprefix Hijack: (1.2.3/24,666) to AS 12~~
- **Downgrade** + Origin Hijack: X=(1.2/16, 666-11)
- **Downgrade** + Path Manipulation: X=(1.2/16, 666-2-11)
- Route Leak (valley): up to AS 3
  - Attribute Manipulation: X=(1.2/16, 666-2-1-11, blackhole) to AS 5

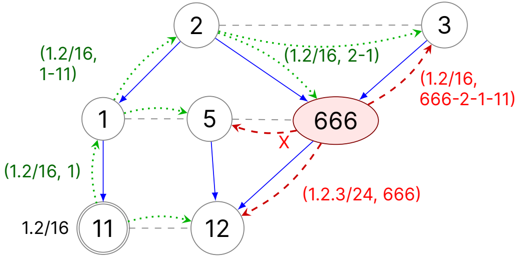


Mis-Routing Attacks in spite of BGPsec (and ROV)

All post-ROV vulnerabilities remain even with global adoption of BGPsec!

- ~~Prefix Hijack: X=(1.2/16, 666) to AS 5~~
- ~~Subprefix Hijack: (1.2.3/24,666) to AS 12~~
- **Downgrade** + Origin Hijack: X=(1.2/16, 666-11)
- **Downgrade** + Path Manipulation: X=(1.2/16, 666-2-11)
- Route Leak (valley): up to AS 3
  - Attribute Manipulation: X=(1.2/16, 666-2-1-11, blackhole) to AS 5

... and BGPsec is also computationally expensive (signatures, verifications)!

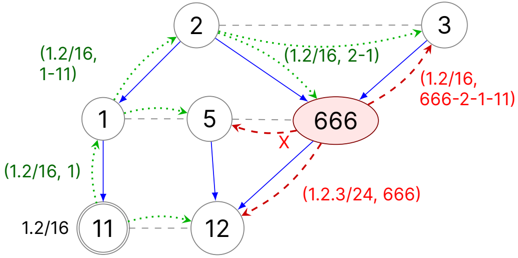


Mis-Routing Attacks in spite of BGPsec (and ROV)

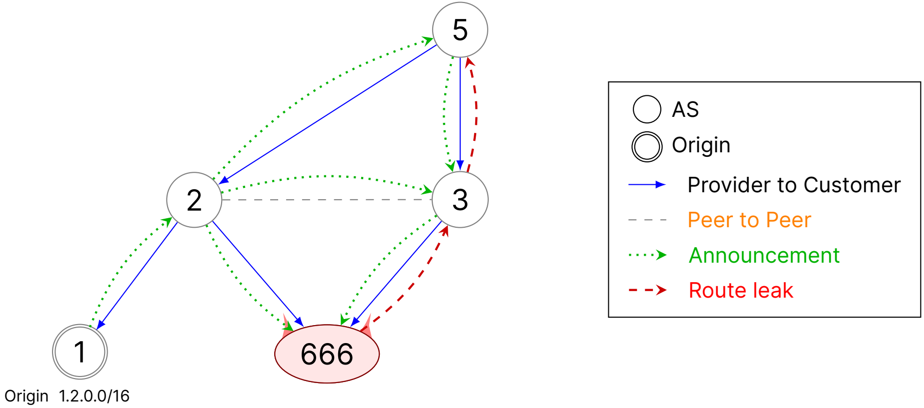
All post-ROV vulnerabilities remain even with global adoption of BGPsec!

- ~~Prefix Hijack: X=(1.2/16, 666) to AS 5~~
- ~~Subprefix Hijack: (1.2.3/24,666) to AS 12~~
- **Downgrade** + Origin Hijack: X=(1.2/16, 666-11)
- **Downgrade** + Path Manipulation: X=(1.2/16, 666-2-11)
- **Route Leak (valley):** up to AS 3
  - Attribute Manipulation: X=(1.2/16, 666-2-1-11, blackhole) to AS 5

... and BGPsec is also computationally expensive (signatures, verifications)!

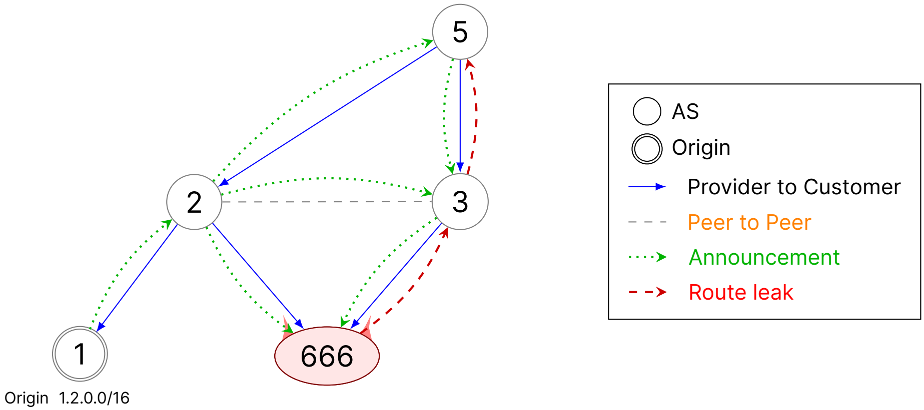


Route leak: export announcement not received from a customer



Announcement from customer should only contain an up-path (customer exports to provider). In announcement from peer, path should be up except the last (peer) edge.

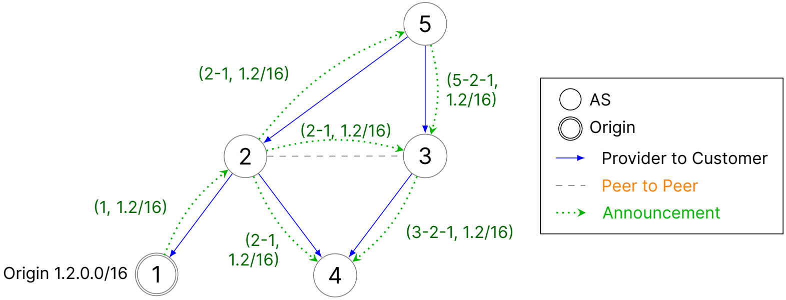
Route leak: export announcement not received from a customer



Leaks can be intentional attacks or not (misconfigurations, ‘fat fingers’)
A leak has a valley, even if it BGP-compliant, e.g.: (666-2-1, 1.2/16) to AS 3

Recall: Valley-Free Routing Policy (Gao-Rexford)

- 1st, prefer routes to maximize profits: Best: from customers (income); 2^{nd} best: from peers (no cost); Worse: from providers ($!!)
  - If same relationship, prefer shorter AS path
- Export customer announcement to all neighbors; if best is from peer/provider, export only to customers.



Defenses against Route Leaks

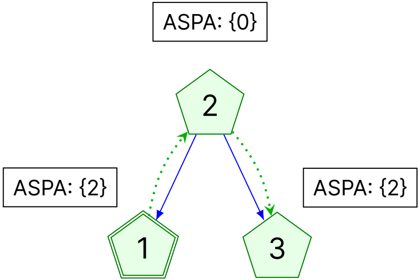
- Prefix and path filtering: only by provider of leaking AS
- Detect and Fightback (announce subprefix): attacker can leak subprefix

Defenses against Route Leaks

- Prefix and path filtering: only by provider of leaking AS
- Detect and Fightback (announce subprefix): attacker can leak subprefix
- **AS Provider Authorization (ASPA)**
- Only-to-Customer (OTC) attribute [RFC9234]
- BGP-iSec route-leak defenses: signed OTC, UP attributes and ProConID

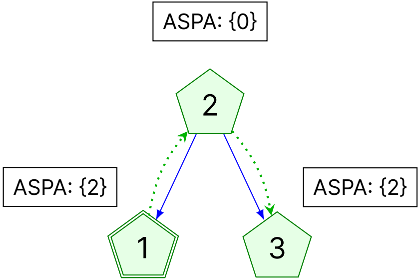
ASPA: AS Provider Authorization

- ASPA: an Internet Draft (I-D) of IETF's SIDR WG; its (main) goal is to foil route leaks.
- ASPA adopting ASes also adopt ROV
- Each AS publishes a Set of Provider ASes
  - ASPA:$\{0\}$ means no provider AS



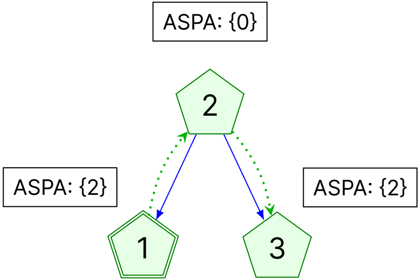
ASPA: AS Provider Authorization

- ASPA: an Internet Draft (I-D) of IETF's SIDR WG; its (main) goal is to foil route leaks.
- ASPA adopting ASes also adopt ROV
- Each AS publishes a Set of Provider ASes
  - ASPA:$\{0\}$ means no provider AS
- Discard announcement if its path contains an adopting AS announcing to a non-provider, followed by adopting AS receiving from non-provider or sending to provider



ASPA: AS Provider Authorization

- ASPA: (main) goal is to foil route leaks.
- Each AS publishes a Set of Provider ASes
- Discard announcement if its path contains an adopting AS announcing to a non-provider, followed by adopting AS receiving from non-provider or sending to provider
- Fully deployed, ASPA ensures (only) path plausibility; can't validate that path was actually announced

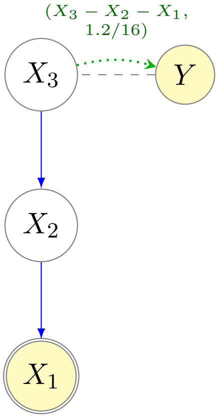


ASPA Validation for announcements from customer/peer

- Customer and bilateral peers should export only announcements from their customers
- $ASPA(X)$: signed list of X's **providers**
  - No providers? $\Rightarrow ASPA(X) = \{AS0\}$
  - $\perp$ if X did not publish ASPA list
- Suppose AS Y receives announcement α from customer/peer X_n, with path: $X_n - \ldots - X_1$
  - Path must be 'upwards': $(\forall 1 < i \leq n) X_{i-1} \notin ASPA(X_i)$ and $(\forall i < n) ASPA(X_i) \neq \perp \Rightarrow X_{i+1} \in ASPA(X_i)$
  - Otherwise: Y discards announcement α

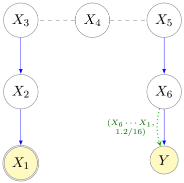
ASPA Validation for announcements from customer/peer

- Customer and bilateral peers should export only announcements from their customers
- $ASPA(X)$: signed list of X's **providers**
  - No providers? $\Rightarrow ASPA(X) = \{AS0\}$
  - $\perp$ if X did not publish ASPA list
- Suppose AS Y receives announcement α from customer/peer X_n, with path: $X_n - \ldots - X_1$
  - Path must be 'upwards': $(\forall 1 < i \leq n) X_{i-1} \notin ASPA(X_i)$ and $(\forall i < n) ASPA(X_i) \neq \perp \Rightarrow X_{i+1} \in ASPA(X_i)$
  - Otherwise: Y discards announcement α
  - In example, Y permits announcement α, iff:
    - $X_2 \in ASPA(X_1)$,
    - X_2 has no ASPA or $X_3 \in ASPA(X_2)$, and
    - $X_2 \notin ASPA(X_3)$



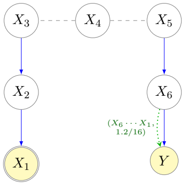
ASPA validation for announcement from provider

- Suppose AS Y receives announcement α from provider X_n, with path: $X_n - \ldots - X_1$
  - Path must be Up*-[Peer]-Down* : $\exists 1 \leq l \leq r \leq n$ s.t.:
    - $(\forall i < l) ASPA(X_i) \neq \perp \Rightarrow X_{i+1} \in ASPA(X_i)$ and $X_{i-1} \notin ASPA(X_i)$,
    - $(\forall i > r) ASPA(X_{i+1}) \neq \perp \Rightarrow X_i \in ASPA(X_{i+1})$ and $X_{i+1} \notin ASPA(X_i)$,
    - $(\forall k(l \leq k < r))$, $X_k \notin ASPA(X_{k+1})$ and $X_{k+1} \notin ASPA(X_k)$,
    - either $r = l + 1$ or $ASPA(X_k) \neq \perp$ for at most one $k \in [l, r]$
  - Otherwise: Y discards announcement α



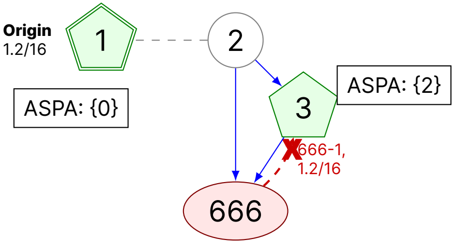
ASPA validation for announcement from provider

- Suppose AS Y receives announcement α from provider X_n, with path: $X_n - \ldots - X_1$
  - Path must be Up*-[Peer]-Down*
  - In example, Y permits α, if:
    - $X_2 \in ASPA(X_1)$; $X_2 \notin ASPA(X_3)$
    - X_2 has no ASPA or $X_3 \in ASPA(X_2)$, and $X_2 \notin ASPA(X_3)$
    - X_6 has no ASPA or $X_5 \in ASPA(X_5)$, and
    - $X_6 \notin ASPA(X_5)$
    - At most one of $\{X_3, X_4, X_5\}$ adopted ASPA
  - Otherwise: Y discards announcement α



ASPA prevents many leaks

- ASPA discards an announcement if its path contains an adopting AS announcing to a non-provider, followed by adopting AS receiving from non-provider or exporting to provider
- This foils many leaks, e.g., 666 leaking with AS-path 666-1 (or 666-2-1)
  - AS 1 has no providers ⇒ leak

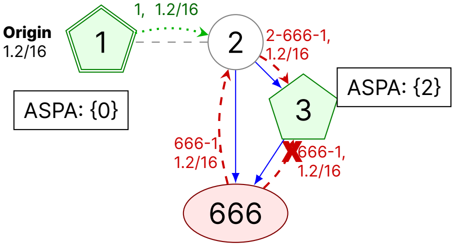


ASPA prevents many (not all!) leaks

- ASPA discards an announcement if its path contains an adopting AS announcing to a non-provider, followed by adopting AS receiving from non-provider or exporting to provider

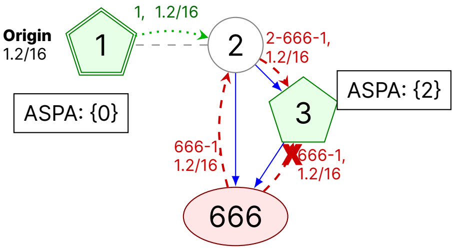
- This foils many leaks, e.g., 666 leaking with AS-path 666-1 (or 666-2-1)
  - AS 1 has no providers ⇒ leak

- But may fail to foil leaks from a rogue or non-adopting **provider**; why?



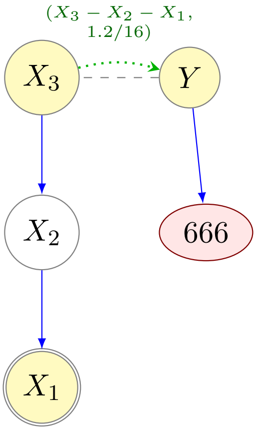
ASPA prevents many (not all!) leaks

- ASPA foils many leaks, e.g., 666 leaking with AS-path 666-1 (or 666-2-1)
  - AS 1 has no providers ⇒ leak
- But may fail to foil leaks from a rogue or non-adopting **provider**; why?
- Many of these can be foiled by an extension called ASRA (AS Relationship Authorization); out of scope



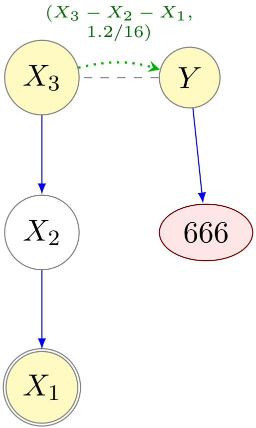
ASPA does not plug all leaks

- ASPA can't prevent a customer from leaking announcement from any non-adopting provider of the origin
  - In example: AS 666 leaks $(666 - X_2 - X_1, 1.2/16)$, an announcement it never received



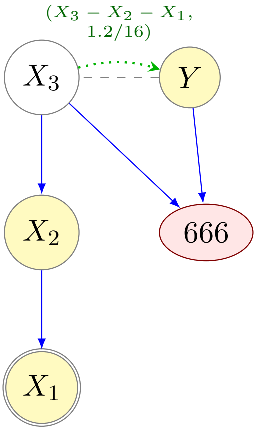
ASPA does not plug all leaks

- ASPA can't prevent a customer from leaking announcement from any non-adopting provider of the origin
  - In example: AS 666 leaks $(666 - X_2 - X_1, 1.2/16)$, an announcement it never received
- ASPA can't prevent an accidental or intentional leak by a customer/peer of a non-adopting AS



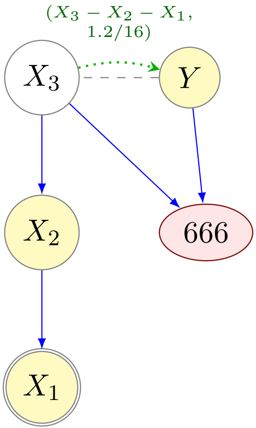
ASPA does not plug all leaks

- ASPA can't prevent a customer from leaking announcement from any non-adopting provider of the origin
- ASPA can't prevent an accidental or intentional leak by a customer/peer of a non-adopting AS
- And ASPA doesn't prevent a leak from propagating to the customer cones of all the peers of a non-adopting AS who received the leak



ASPA does not plug all leaks

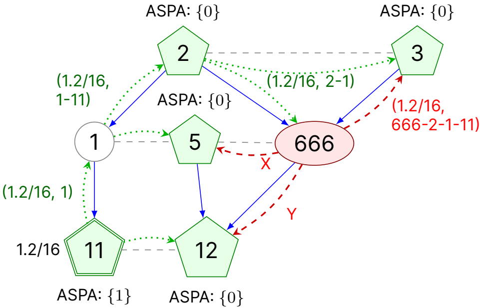
- ASPA can't prevent a customer from leaking announcement from any non-adopting provider of the origin
- ASPA can't prevent an accidental or intentional leak by a customer/peer of a non-adopting AS
- And ASPA doesn't prevent a leak from propagating to the customer cones of all the peers of a non-adopting AS who received the leak
- Many of these leaks are prevented by ASRA (Autonomous System Relationship Authorization) - out of scope



Mis-Routing Attacks in spite of **ASPA** (and ROV)

ASPA prevents many route leaks and some other attacks

- ~~Prefix Hijack: X=(1.2/16, 666) to AS 5~~
- ~~Subprefix Hijack: Y=(1.2.3/24,666) to AS 12~~
- **Foils many leaks**, e.g., up to AS 3 (if ASes 2,3 adopts)
- **Foils** origin hijack **to non-customer**: X=(1.2/16, 666-11) (if 11, 5 adopt)
- **Foils some** path manipulations: X=(1.2/16, 666-2-11) (if 11, 5 adopt)

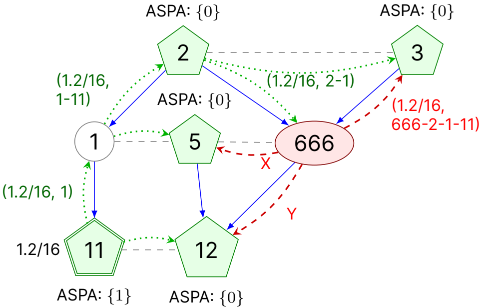


UCONN SCHOOL OF ENGINEERING

Mis-Routing Attacks in spite of **ASPA** (and ROV)

ASPA prevents many route leaks and some other attacks

- ~~Prefix Hijack: X=(1.2/16, 666) to AS 5~~
- ~~Subprefix Hijack: Y=(1.2.3/24,666) to AS 12~~
- **Foils many leaks**, e.g., up to AS 3 (if ASes 2,3 adopts)
- **Foils** origin hijack **to non-customer**: X=(1.2/16, 666-11) (if 11, 5 adopt)
- **Foils some** path manipulations: X=(1.2/16, 666-2-11) (if 11, 5 adopt)
- **But not all**, e.g, X=(1.2/16, 666-1-11), Y=(1.2/16, 666-11)
  - Also not attribute manipulation: Y=(1.2/16, 666-2-1-11, blackhole)



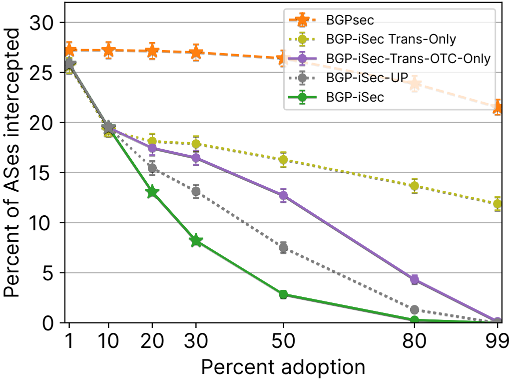
BGP-iSec: **improved** security for BGP

BGP-iSec aims to improve on the security of BGPsec, esp. in partial adoption, with few modifications to the BGPsec design. The main modifications:

- Enable partial path verification.
- Identify adopters and their PK, prevent unauthorized downgrades to BGP.
- Authenticate integrity-protected attributes.
- Effective defenses against route leaks (better than ASPA).

BGP-iSec Components

- Path integrity defense: transitive Signatures.
- Route-leak defenses:
    - Signed OTC attribute.
    - Up-Permitted attributes.
    - ProConID mechanism.



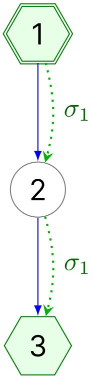
Transitive Signatures (1/2)

- Signatures in BGPsec have the transitive bit set to **false**. They are not sent to BGP neighbors that do not run BPGsec.

- Signatures in Secure BGP (S-BGP, [Kent et al., 2000]) had the transitive bit set to **true**, but they were not sent to neighbors who were not running S-BGP.

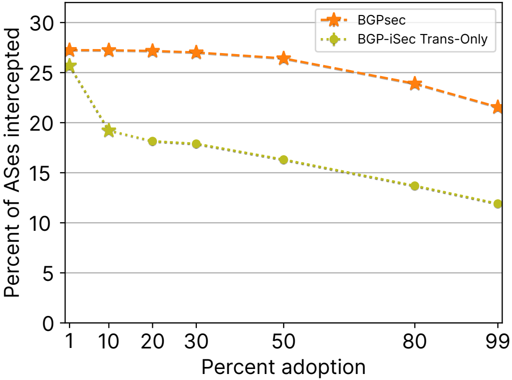
- BGP-iSec sets the transitive bit to **true** and *sends signatures to non-adopting neighbors*.

- Transitive signatures allow BGP-iSec to *enforce downgrade prevention* and *authenticate adopting (sub)paths*.



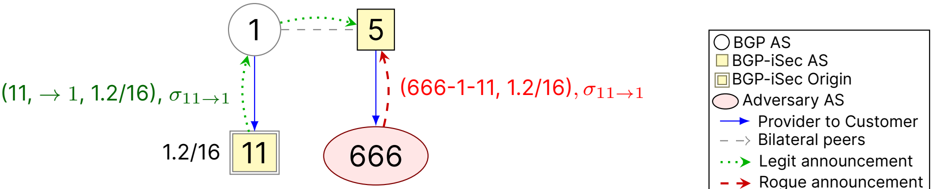
Transitive Signatures (2/2)

- BGP-iSec prevents fake downgrades: signatures are relayed by all ASes; RPKI identifies adopters, keys
- Significant security - for some overhead
- Transitive signatures with partial path verification alone completely prevent hijacks of adopting origins.
- Protects announcement integrity, e.g., the OTC anti-leakage mechanism.



Transitive signatures don't prevent BGP-compliant leaks

- Route leak: attacker exports to provider/peer a path it did not receive from a customer
- With BGPsec, attacker can (leak) origin-hijack by degrading to BGP
- With transitive signatures (BGP-iSec), attacker can leak announcement they received or hijack from a non-adopting AS, e.g., AS 1.
- Rogue announcement, but contains a valid signatures by adopting ASes!

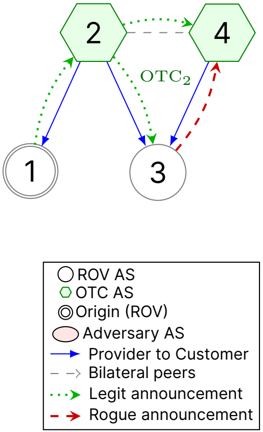


Defenses against Route Leaks

- Prefix and path filtering: only by provider of leaking AS
- Detect and Fightback (announce subprefix): attacker can leak subprefix
- AS Provider Authorization (ASPA)
- **Only-to-Customer (OTC) attribute** [RFC9234]
- BGP-iSec route-leak defenses: signed OTC, UP attributes and ProConID

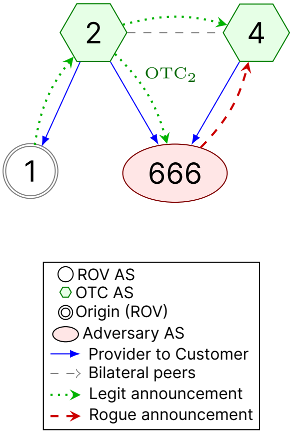
Only-To-Customer (OTC) (1/2)

- RFC-9234 defines the OTC attribute, indicating that the route should be propagated Only To Customers.
  - Adopting AS sets when sending to customer/peer; drops announcement with OTC if received from customer, also from peer except the peer who set OTC
  - In example, allows AS 4 to ignore leak from AS 3
- OTC prevents unintentional leaks; growing adoption.



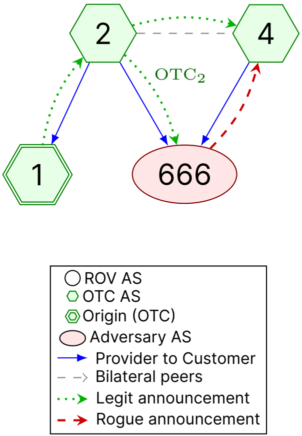
Only-To-Customer (OTC) (1/2)

- RFC-9234 defines the OTC attribute, indicating that the route should be propagated Only To Customers.
- OTC prevents unintentional leaks; growing adoption.
- The OTC attribute is unauthenticated; a malicious attacker can remove it.
  - AS 666 removes OTC_2, causing AS 4 to route via AS 666



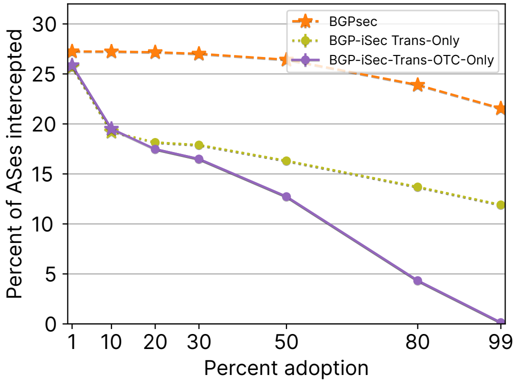
Signed Only-To-Customer (OTC) (1/2)

- RFC-9234 defines the OTC attribute, indicating that the route should be propagated Only To Customers.
- OTC prevents unintentional leaks; growing adoption.
- The OTC attribute is unauthenticated; a malicious attacker can remove it.
- BGP-iSec' transitive signatures authenticate the OTC attribute, preventing also malicious route leaks.
  - If AS 666 removes OTC_2, AS 4 discards rogue announcement since it will not be well-signed by AS 2
  - AS 1 should also adopt BGP-iSec, otherwise, AS 666 can origin hijack instead



Signed Only-To-Customer (OTC) (2/2)

- By authenticating OTC [RFC9234], BGP-iSec foils significantly more post-ROV routing attacks.
- OTC attributes are already in use; authenticates on reaching BGP-iSec adopting AS.
- BGP-iSec has two other defenses which improve prevention of intentional leaks: the UP attributes and the ProConID mechanism



Defenses against Route Leaks

- Prefix and path filtering: only by provider of leaking AS
- Detect and Fightback (announce subprefix): attacker can leak subprefix
- AS Provider Authorization (ASPA)
- Only-to-Customer (OTC) attribute [RFC9234]
- BGP-iSec route-leak defenses: signed OTC, UP attributes and ProConID

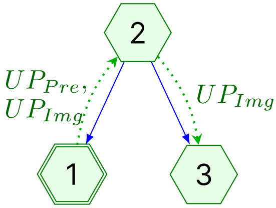
BGP-iSec UP (Up Permitted) Attributes (1/2)

- The two *Up-Permitted* (UP) attributes, UP_{Pre} and UP_{Img}, indicate whether an announcement can be sent to providers (upward).

- UP_{Pre} contains a random string x; UP_{Img} contains $h(x)$, where h is a crypto-hash function

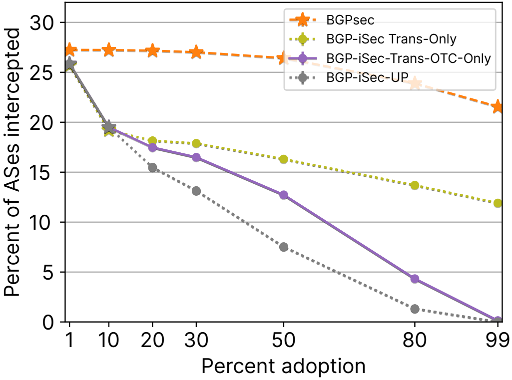
- The UP Preimage is removed when an announcement is sent to a customer or peer (downward).

- Since the hash function cannot be reversed, the preimage cannot be re-added.



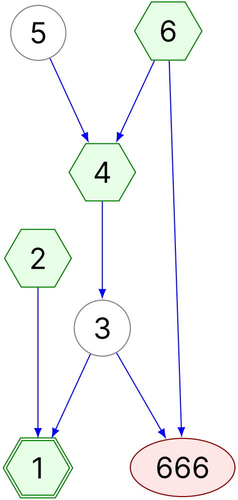
BGP-iSec Up Permitted (UP) Attributes (2/2)

- Authenticated UP attributes make shortening a leaked AS path more difficult.
- Hash functions are computationally efficient and the digests can be small.
- Drawback: an eavesdropping adversary can capture the preimage.



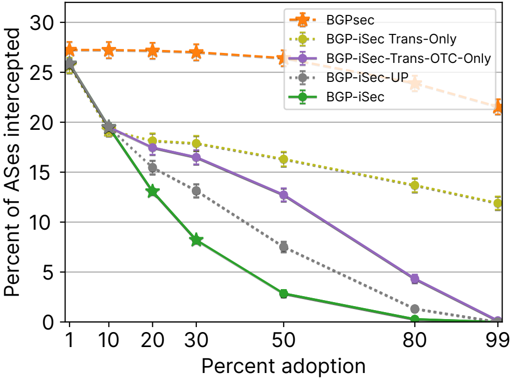
BGP-iSec ProConID (1/2)

- Adopting AS X signs P_X, a list of X's nearest-provider BGP-iSec ASes. E.g: $P_1 = \{2, 4\},\ P_4 = \{6\}$ and P_2 is empty.
- Let $\{X_i\}_{i=1}^{n}$ be the adopting ASes in announcement α received by X_n. If $(\exists i) X_i \notin P_{X_{i+1}}$ then X_n drops α.
- E.g., AS 6 only allows announcements whose path contains $\{1, 4, 6\}$; e.g., it drops path $(666 - 3 - 1)$. And $(666 - 5 - 4 - 3 - 1)$ is too long.



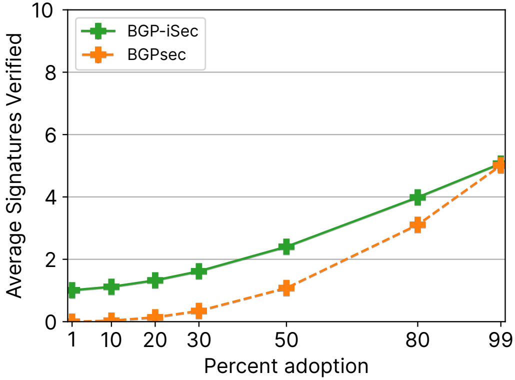
BGP-iSec ProConID (2/2)

- ProConID provides even stronger protection against route leaks than UP attributes.
- Provider cones are small on average (median size is around 30).
- The overhead of updating and maintaining the ProConID-list is reasonably low (see in paper).



Overhead Comparison: BGPsec vs. BGP-iSec

- Both BGPsec and BGP-iSec require the same number of signature verification operations in full deployment.
- More signatures on average are verified in partial adoption because they transit over non-adopting ASes.
- In BGPsec, signatures are limited to deployment “islands”.



Conclusions

- BGP security is challenging
- Many autonomous systems (ASes), conflicting interests, may 'break rules'
- New, improved defenses: ROV, ROV++, BGPsec, BGP-iSec, OTC, ASPA....
- Challenges: partial deployment and incentives
- Some of the many topics not (yet?) covered in this presentation:
    - Exploiting routing attacks: de-anonymization (TOR), DNS-poisoning, defeating domain-validation (get misleading certificates), email interception, ...
    - Routing security aware defenses in applications
    - Source address validation (SAV): uRPF and beyond
    - Data plane failures and attacks: DoS, failure to ensure QoS, and more

Thank you for your attention

Questions?

amir.herzberg@gmail.com

Amir Herzberg, University of Connecticut

Vielen Dank für Ihre Aufmerksamkeit!

Amir Herzberg, University of Connecticut

Fragen? (Bitte, in English)

Backup

Simulation[2]-based Evaluation of BGP-iSec

Assumptions:

- Post-ROV: ROA for prefixes, ROV by all ASes
- Valley-free Routing (with export-to-all)
- Relationships (topology) from CAIDA [serial 2]
- Identified Adopters and Public Keys (e.g. in RPKI)
- Security Third
  - If two received paths are from same type of neighbor (e.g., provider) and have same length, prefer the fully-adopting one

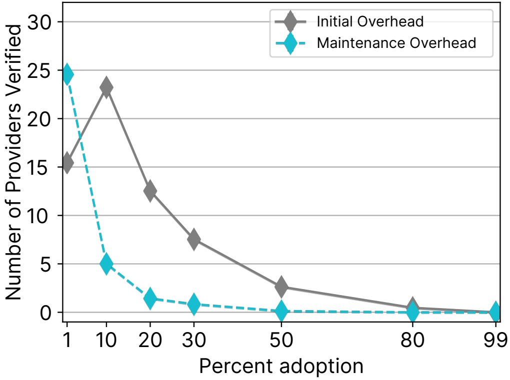
[2] Simulations were performed using custom extensions to BGPy https://github.com/jfuruness/bgpy_pkg

Evaluation: Attacker Models

- **Full Attacker**: Receives all BGP announcements sent by every AS including BGP-iSec attributes.
- **Global Attacker**: Receives all BGP announcements sent by every AS, but does not receive BGP-iSec attributes.

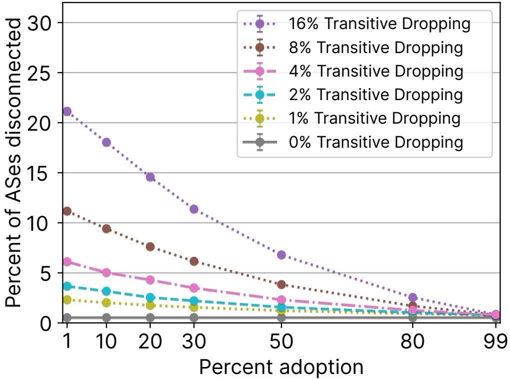
Overhead of ProConID

- ProConID requires confirming the set of ASes in one's provider cone.
- Initial overhead shows the average number of providers verified when an AS first adopts ProConID.
- Maintenance overhead reflects additional providers they need to verify are in their provider cone as adoption increases.



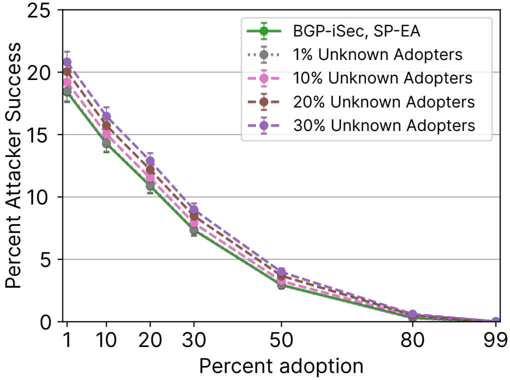
Dropped Transitive Attributes?

- Almost all (98-99% of) BGP routers forward transitive attributes they do not recognize, but this behavior is a “SHOULD” requirement in the RFC.
- A dropped transitive signature is indistinguishable from a downgrade attack.
- An AS should ensure its neighbors do not drop unrecognized transitive attributes before enforcing transitive signatures.



Unknown Adopters?

- So far, we assumed BGP-iSec adopters and their public keys would be known to other adopters, via the RPKI or some other mechanism.
- The overall impact of even a large number of unknown adopters is small.



UCONN SCHOOL OF ENGINEERING