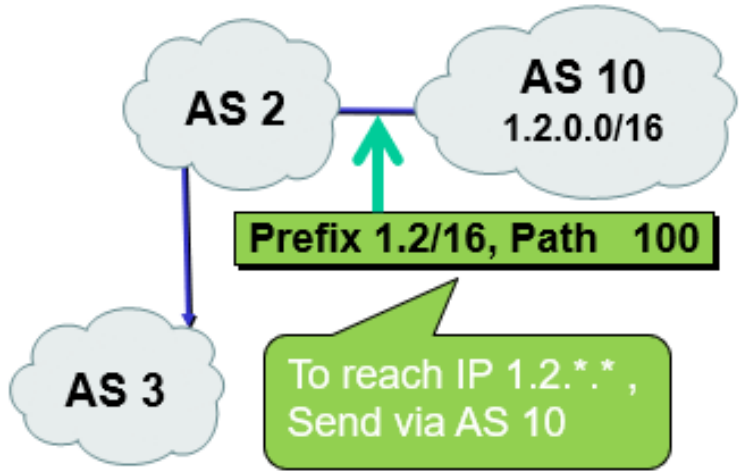
DISCO: Sidestepping RPKI's Deployment Barriers

Tomas Hlavacek[1] Italo Cunha[23] Yossi Gilad[4] Amir Herzberg[5]
Ethan Katz-Bassett[3] Michael Schapira[4] Haya Shulman[1]

[1]Fraunhofer SIT [2]Universidade Federal de Minas Gerais [3]Columbia University
[4]Hebrew University of Jerusalem [5]University of Connecticut

The Border Gateway Protocol (BGP)

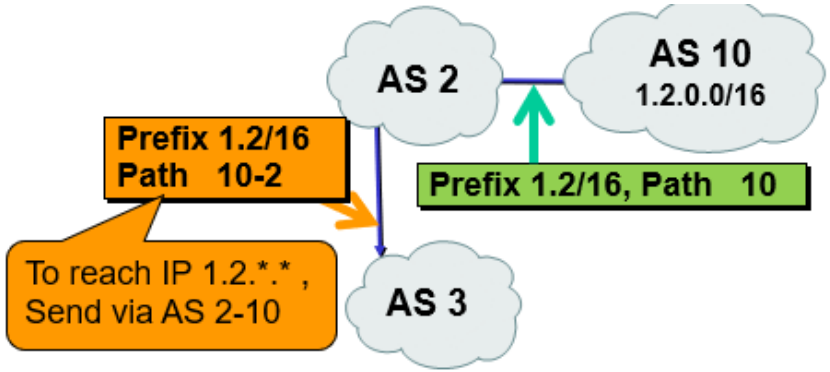
- The Internet is composed of many Autonomous Systems (ASes)
  - Aka ISPs or Domains
- Inter-AS routing uses **BGP**
- Example: AS 10 announces it has prefix 1.2.0.0/16 to AS 2



Inter-AS routing with BGP: AS 10 announces prefix 1.2.0.0/16 to AS 2.

The Border Gateway Protocol (BGP)

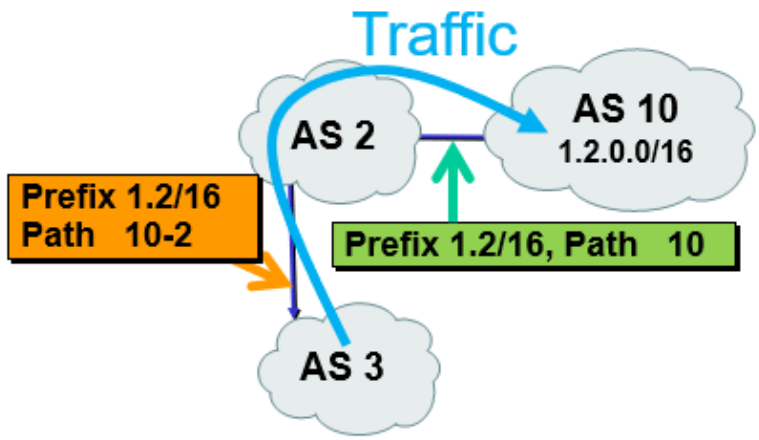
- The Internet is composed of many Autonomous Systems (ASes)
  - Aka ISPs or Domains
- Inter-AS routing uses **BGP**
- Example: AS 10 announces it has prefix 1.2.0.0/16 to AS 2
- AS 2 forwards to AS 3



Inter-AS routing with BGP: AS 10 announces prefix 1.2.0.0/16 to AS 2, who forwards to AS 3.

The Border Gateway Protocol (BGP)

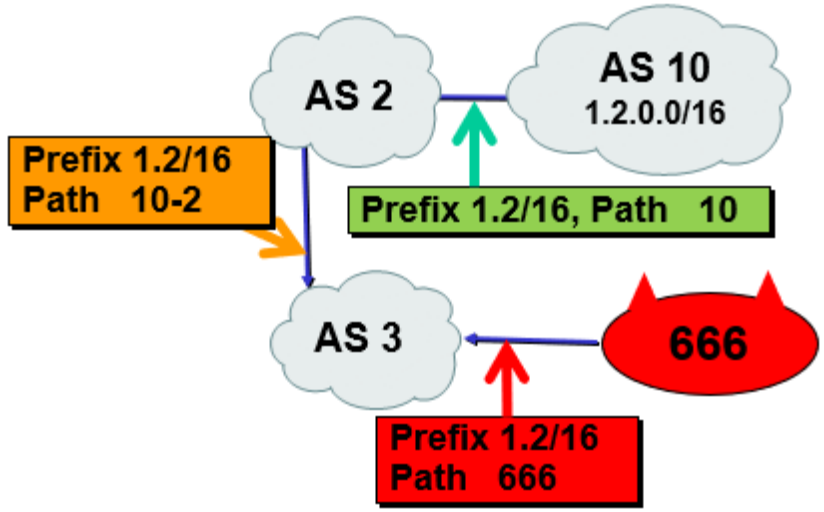
- The Internet is composed of many Autonomous Systems (ASes)
  - Aka ISPs or Domains
- Inter-AS routing uses **BGP**
- Example: AS 10 announces it has prefix 1.2.0.0/16 to AS 2
- AS 2 forwards to AS 3
- AS 3 routes to 1.2/16 via AS 2



Inter-AS routing with BGP: AS 10 announces prefix 1.2.0.0/16 to AS 2, who forwards to AS 3. Now AS 3 sends traffic to 1.2/16 (via AS 2).

Internet Inter-Domain Routing (In)Security

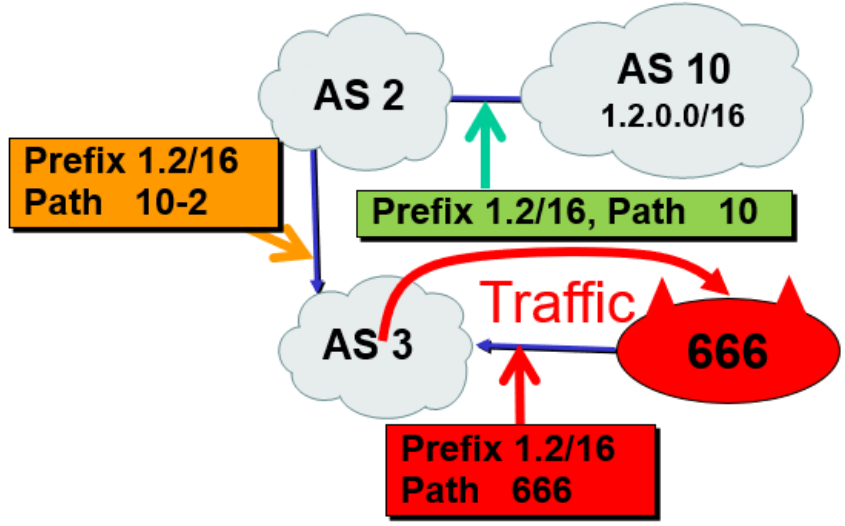
- BGP has no built-in security mechanism
- Long history of attacks and problems:
  - route manipulations, mostly prefix hijacks
  - route leaks
  - intentional and benign - but always painful...
- Example of prefix hijack: AS 666 claims to host 1.2.0.0/16.



AS 666 announces prefix 1.2.0.0/16 to AS 3.

Internet Inter-Domain Routing (In)Security

- BGP has no built-in security mechanism
- Long history of attacks and problems:
  - route manipulations, mostly prefix hijacks
  - route leaks
  - intentional and benign - but always painful...



AS 666 announces prefix 1.2.0.0/16 to AS 3. AS 3 sends traffic to 666 (e.g., shorter path)

Internet Inter-Domain Routing (In)Security

- BGP has no built-in security mechanism
- Long history of attacks and problems:
  - route manipulations, mostly prefix hijacks
  - route leaks
  - intentional and benign - but always painful...
- Defenses? ad-hod, proprietary (expensive), weak

Internet Inter-Domain Routing (In)Security

- BGP has no built-in security mechanism
- Long history of attacks and problems:
  - route manipulations, mostly prefix hijacks
  - route leaks
  - intentional and benign - but always painful...
- Defenses? ad-hod, proprietary (expensive), weak
- BGPsec (RFCs published in 2017)
  - Ambitious: prevent all route manipulations

Internet Inter-Domain Routing (In)Security

- BGP has no built-in security mechanism
- Long history of attacks and problems:
  - route manipulations, mostly prefix hijacks
  - route leaks
  - intentional and benign - but always painful...
- Defenses? ad-hod, proprietary (expensive), weak
- BGPsec (RFCs published in 2017)
  - Ambitious: prevent all route manipulations
  - But deployment is hard/unlikely
  - And: builds on RPKI...
- RPKI (RFCs published in 2012)
  - (only) prevent prefix hijacks
  - **Our focus**

RPKI: Resource Public Key Infrastructure

- Routing Certificate (RC): binds IP prefix π to public key *pk*
- Route Origin Authorization (ROA): binds (prefix,origin) pair
  - *Max-Length*: most-specific subprefix allowed
  - Signed by public key *pk* (certified for π)
- Route Origin Validation (ROV): validate origin in BGP announcements
  - Deployed by BGP routers
  - Discard announcement with 'invalid' (prefix,origin) pair (differ from ROA)

RPKI: Resource Public Key Infrastructure

- Routing Certificate (RC): binds IP prefix π to public key *pk*
- Route Origin Authorization (ROA): binds (prefix,origin) pair
  - *Max-Length*: most-specific subprefix allowed
  - Signed by public key *pk* (certified for π)
- Route Origin Validation (ROV): validate origin in BGP announcements
  - Deployed by BGP routers
  - Discard announcement with 'invalid' (prefix,origin) pair (differ from ROA)
  - 18.5% of (prefix,origin) pairs are 'valid', 0.8% 'invalid' [NIST]
    - Others (81.7%): no ROA
  - Concern: most 'invalid' due to 'wrong' ROA, not to hijack
  - Limited security benefits - esp. for partial adoption
  - $\Rightarrow$ Slow adoption

Research on Deploying RPKI

- RPKI ecosystem and deployment:
  Wahlisch*CCR12, Iamartino*PAM15, Wahlisch*HotNet15,
  Gilad*NDSS17, Gilad*HotNts18, Reuters*CCR18,
  Hlavacek*DSN18, Chung*IMC19, Testart*PAM20
- RPKI security concerns, extensions:
  - Misbehaving authority: Cooper*HotNts13, Heilman*SigCom14
  - 'Path-end' extension: Cohen*SigComm16
  - Max-Length considered harmful: Gilad*CoNext17

Research on Deploying RPKI

- RPKI ecosystem and deployment:
  Wahlisch*CCR12, Iamartino*PAM15, Wahlisch*HotNet15,
  Gilad*NDSS17, Gilad*HotNts18, Reuters*CCR18,
  Hlavacek*DSN18, Chung*IMC19, Testart*PAM20
- RPKI security concerns, extensions:
  - Misbehaving authority: Cooper*HotNts13, Heilman*SigCom14
  - 'Path-end' extension: Cohen*SigComm16
  - Max-Length considered harmful: Gilad*CoNext17
- This work (DISCO):
  - Complementary, automated Routing Certification mechanism
  - Goal: easy-to-issue and correct ROAs, RCs

Pitfalls with RPKI Issuing of RCs, ROAs

- Routing Certificates (RCs):
  - Manual application by Origin-AS network manager
    - Errors have legal/business implications!
  - Room for errors, e.g., forgotten/wrong prefix, origin-AS
  - No (immediate) feedback on errors
  - Validation: manual - based on records of assignment, transfer
- Route Origin Authorizations (ROAs):
  - Manual issuing by Origin-AS network manager
    - Errors have legal/business implications!
  - Large space for errors
    - Forgotten prefix/originAS/subprefix, wrong/missing Max-Length,...
  - No validation, no (immediate) feedback on errors

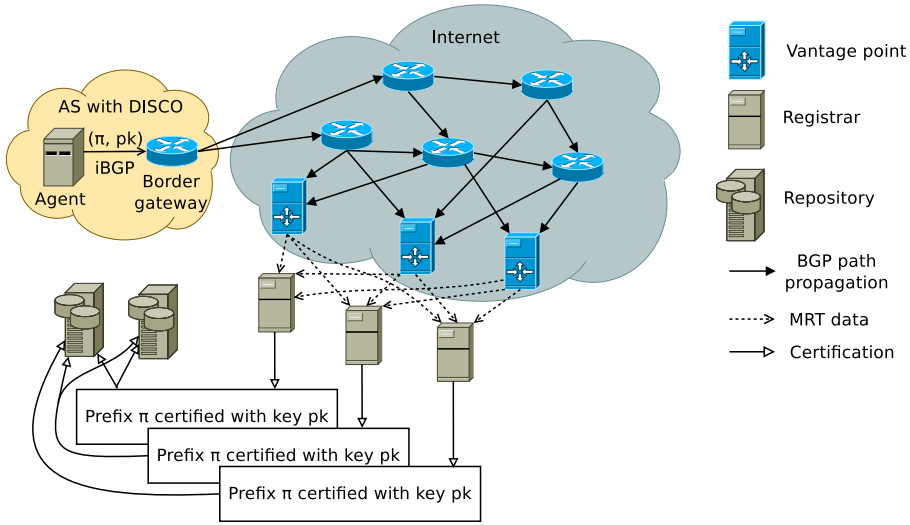
Pitfalls with RPKI Issuing of RCs, ROAs

- Routing Certificates (RCs):
  - Manual application by Origin-AS network manager
    - Errors have legal/business implications!
  - Room for errors, e.g., forgotten/wrong prefix, origin-AS
  - No (immediate) feedback on errors
  - Validation: manual - based on records of assignment, transfer
- Route Origin Authorizations (ROAs):
  - Manual issuing by Origin-AS network manager
    - Errors have legal/business implications!
  - Large space for errors
    - Forgotten prefix/originAS/subprefix, wrong/missing Max-Length,...
  - No validation, no (immediate) feedback on errors
- Like Waltz: great - if done well... But few do it (right)!
- **Let's DISCO:** easier, and: 'fool-proof'

DISCO

Decentralized Infrastructure for Securing & Certifying Origins

- Automated to reduce errors, ease adoption
  - Let's focus on issuing of Route Certificate (RC)
    - ROAs: later
  - DISCO-agent distributes (prefix π, pk) via BGP
  - Registrar-agents (1) validate, (2) certify and send to repositories
    - Details: next

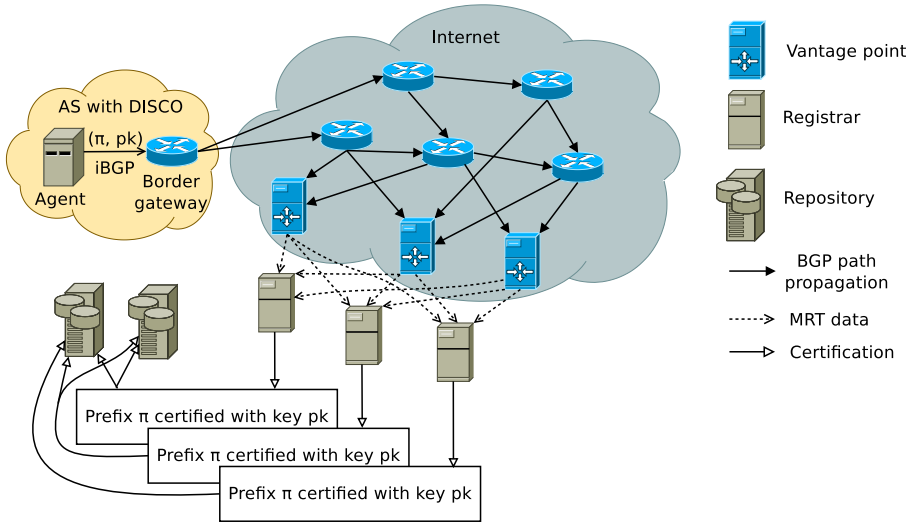


DISCO: automated issuing of RC for prefix π. DISCO registrars validate the (π, pk) pair sent by agent.

DISCO

Decentralized Infrastructure for Securing & Certifying Origins

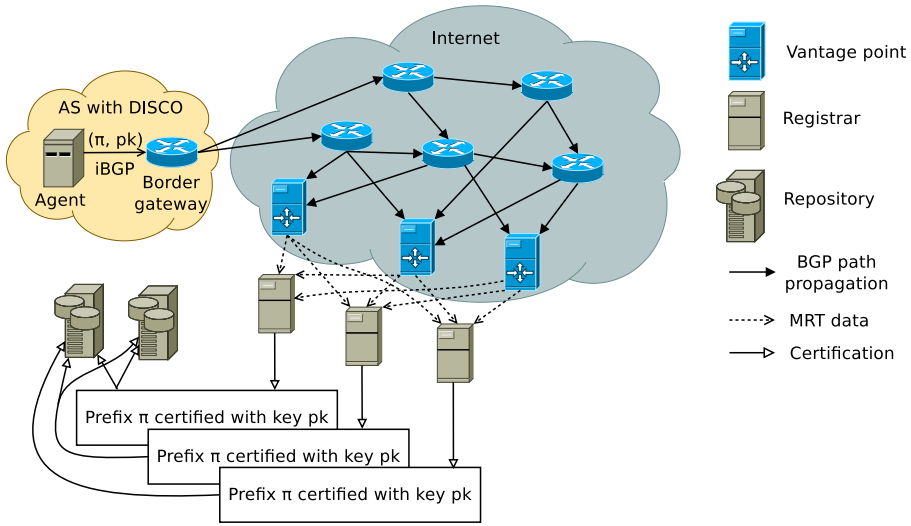
- Automated to reduce errors, ease adoption
  - Let's focus on issuing of Route Certificate (RC)
    - ROAs: later
  - DISCO-agent distributes (prefix π, pk) via BGP
  - Registrar-agents (1) validate, (2) certify and send to repositories
    - Details: next
  - DISCO RCs complement RPKI RCs
    - Conflict handling TBD



DISCO: automated issuing of RC for prefix π. DISCO registrars validate the (π, pk) pair sent by agent.

DISCO: (1) automated validation of (pk,π) to issue RC

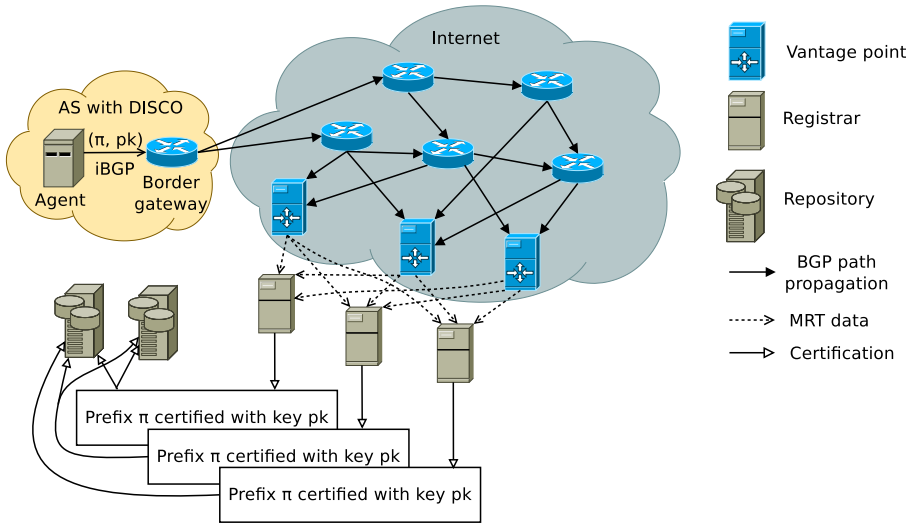
- DISCO-agent announces prefix π, via iBGP, as optional transitive attribute
  - RFC: should relay such attributes
  - Experiments: relayed by almost all ASes



DISCO: automated issuing of RCs. DISCO Registrars validate the (π, pk) pair sent by agent. Agent encodes pk in transitive attribute.

DISCO: (1) automated validation of (pk,π) to issue RC

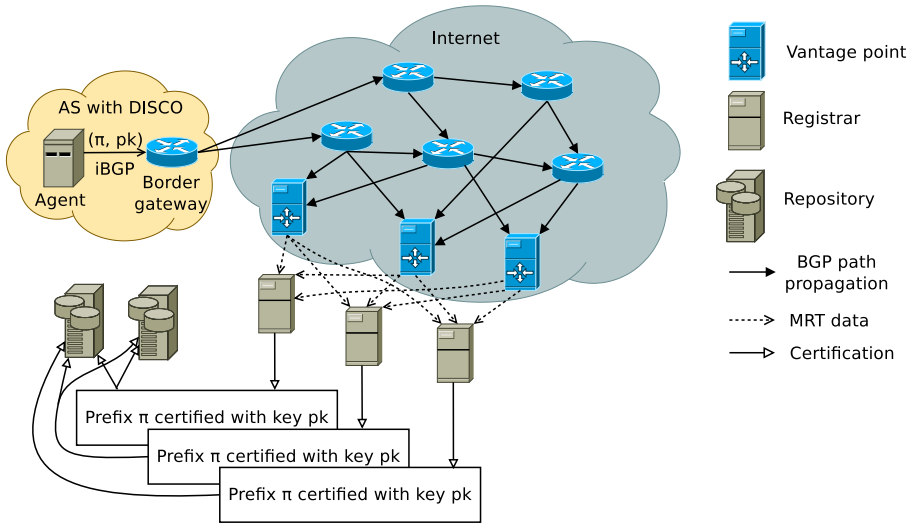
- DISCO-agent announces prefix π, via iBGP, as optional transitive attribute
  - RFC: should relay such attributes
  - Experiments: relayed by almost all ASes
- Registrars validate same pk received from (most) announcements of π
  - Same or different origin AS



DISCO: automated issuing of RCs. DISCO Registrars validate the (π, pk) pair sent by agent. Agent encodes pk in transitive attribute.

DISCO: (1) automated validation of (pk,π) to issue RC

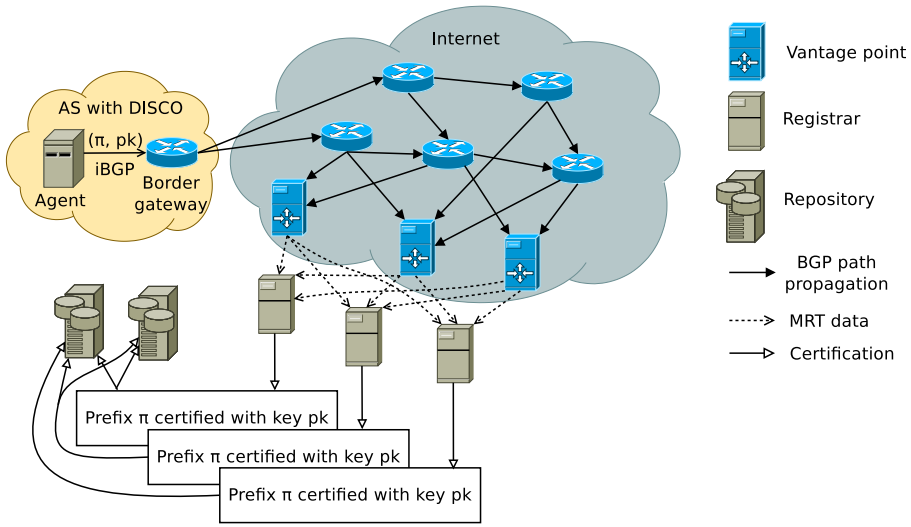
- DISCO-agent announces prefix π, via iBGP, as optional transitive attribute
  - RFC: should relay such attributes
  - Experiments: relayed by almost all ASes
- Registrars validate same pk received from (most) announcements of π
  - Same or different origin AS
- Works for $\geq$ 97% of prefixes
  - N/A for un-announced prefixes, multi-home ($< 1\%$)



DISCO: automated issuing of RCs. DISCO Registrars validate the (π, pk) pair sent by agent. Agent encodes pk in transitive attribute.

DISCO: (2) automated issuing, distributing RC (after validation)

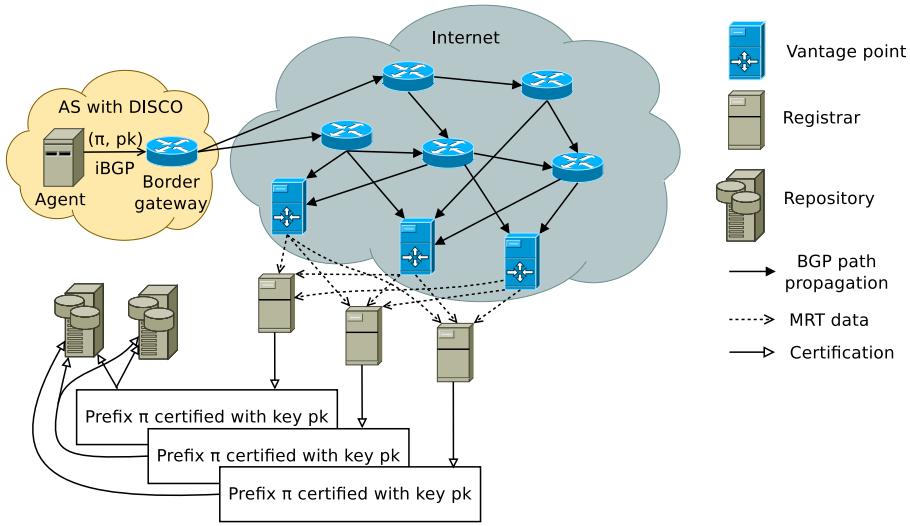
- Each DISCO registrar R_i has a share of threshold signing-key s_i
- Registrar R_i uses share s_i to partially-sign (pk,π) pair, and sends to repository



DISCO: automated issuing of RCs. Registrar R_i validates the (π, pk) pair, then partially-signs it and sends to repositories. Repositories combine partial-signatures to create RC for π.

DISCO: (2) automated issuing, distributing RC (after validation)

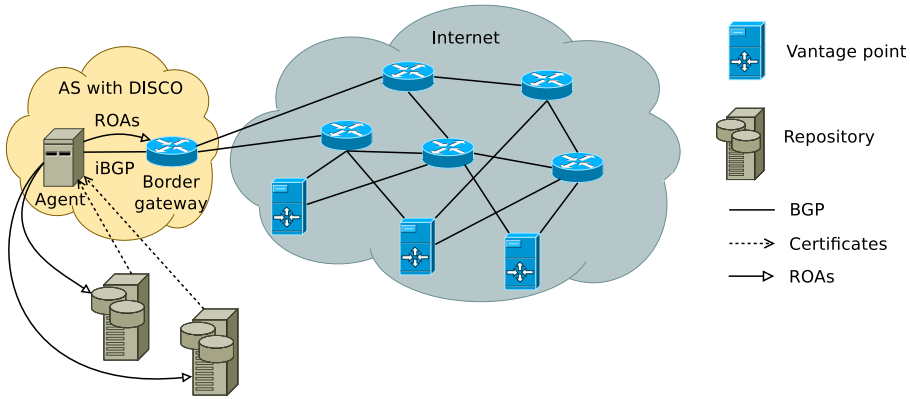
- Each DISCO registrar R_i has a share of threshold signing-key s_i
- Registrar R_i uses share s_i to partially-sign (pk,π) pair, and sends to repository
- Repositories combine partial-signatures and issue RC, i.e. certified (pk, π)
- Resiliency and security by redundancy of paths, registries and repositories
- Repositories provide both DISCO-RCs and RPKI-RCs



DISCO: automated issuing of RCs. Registrar R_i validates the (π, pk) pair, then partially-signs it and sends to repositories. Repositories combine partial-signatures to create RC for π.

DISCO: (3) Issuing ROAs

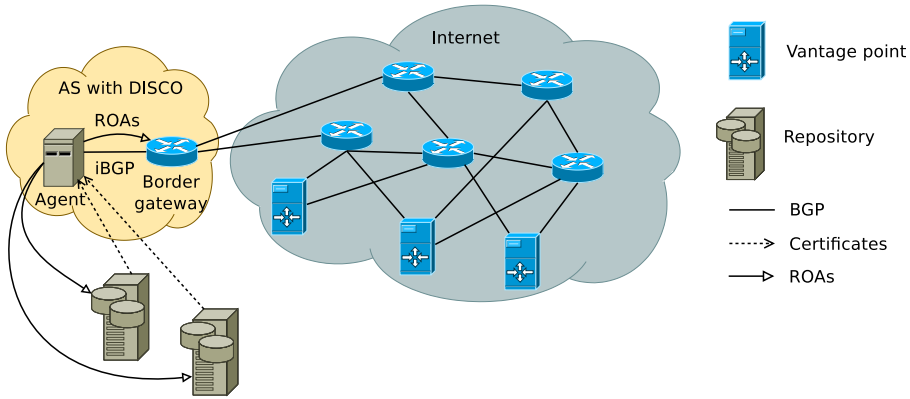
- ROA automatically issued by DISCO-agent
- Agent detects RC was certified and is in repositories
- Agent signs ROA for each (sub)prefix announced by AS



DISCO: automated issuing of correct ROAs to all announced (sub)prefixes. Max-Length used if more efficient (and then for all subprefixes).

DISCO: (3) Issuing ROAs

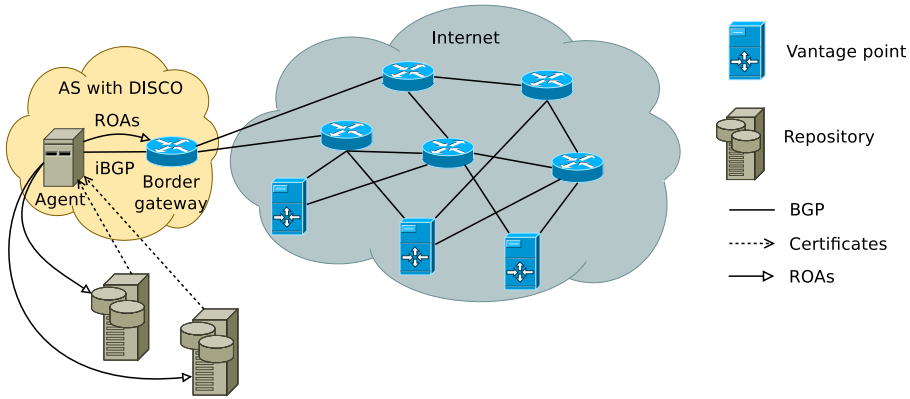
- ROA automatically issued by DISCO-agent
- Agent detects RC was certified and is in repositories
- Agent signs ROA for each (sub)prefix announced by AS
  - Max-Length: only for *all* subprefixes



DISCO: automated issuing of correct ROAs to all announced (sub)prefixes. Max-Length used if more efficient (and then for all subprefixes).

DISCO: (3) Issuing ROAs

- ROA automatically issued by DISCO-agent
- Agent detects RC was certified and is in repositories
- Agent signs ROA for each (sub)prefix announced by AS
  - Max-Length: only for *all* subprefixes

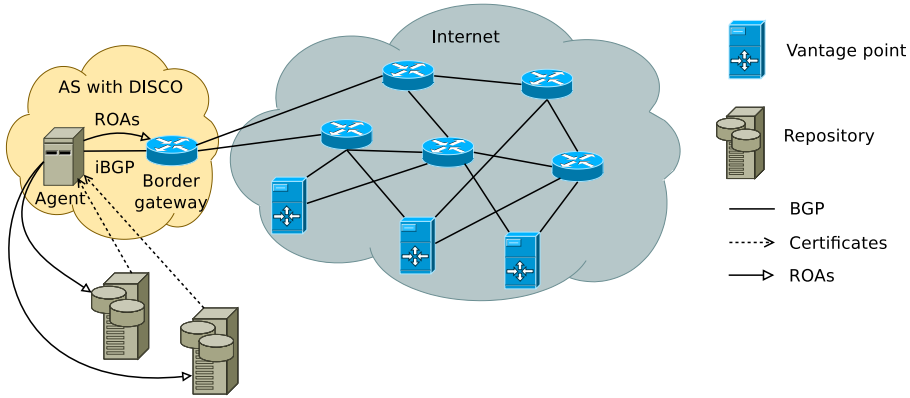


DISCO: automated issuing of correct ROAs to all announced (sub)prefixes. Max-Length used if more efficient (and then for all subprefixes).

- AS 0: un-announced subprefix
- AS *: unprotected subprefix (!!)

DISCO: (3) Issuing ROAs

- ROA automatically issued by DISCO-agent
- Agent detects RC was certified and is in repositories
- Agent signs ROA for each (sub)prefix announced by AS
  - Max-Length: only for *all* subprefixes
  - Automated - or semi-automated, for off-line signing key
- Exchange ROAs with repositories, routers

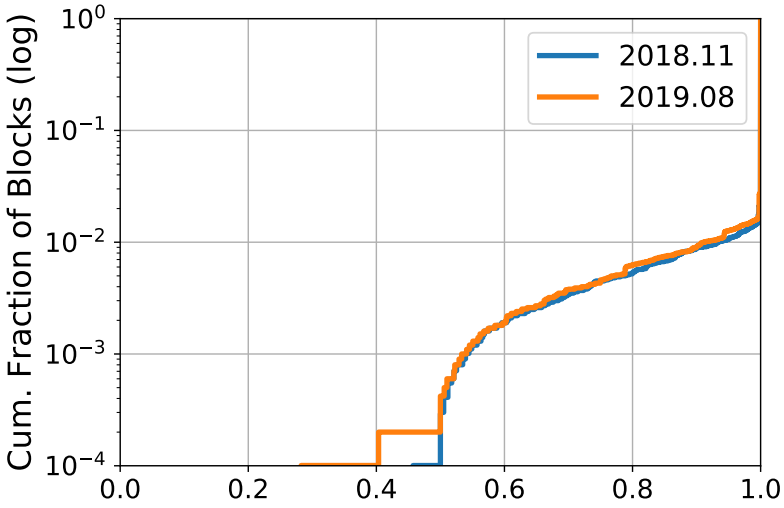


DISCO: automated issuing of correct ROAs to all announced (sub)prefixes. Max-Length used if more efficient (and then for all subprefixes).

- AS 0: un-announced subprefix
- AS *: unprotected subprefix (!!)

DISCO: Experimental Evaluation

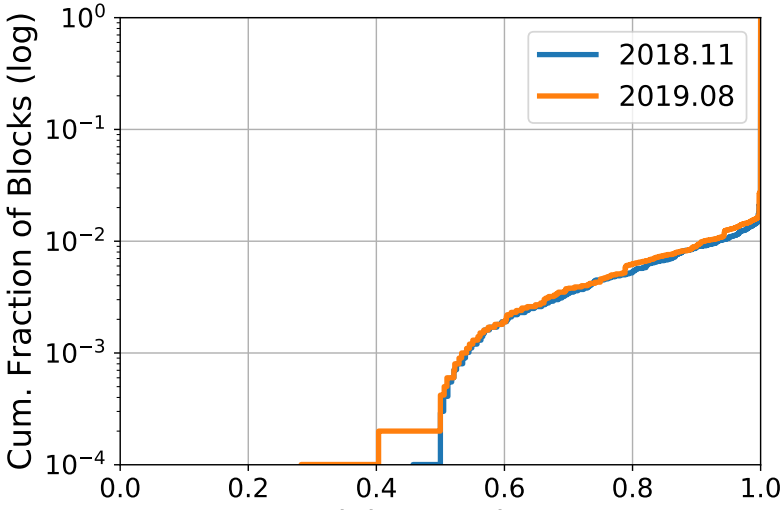
- PK sent via Transitive Attribute `0xff`
  - reserved for testing and development



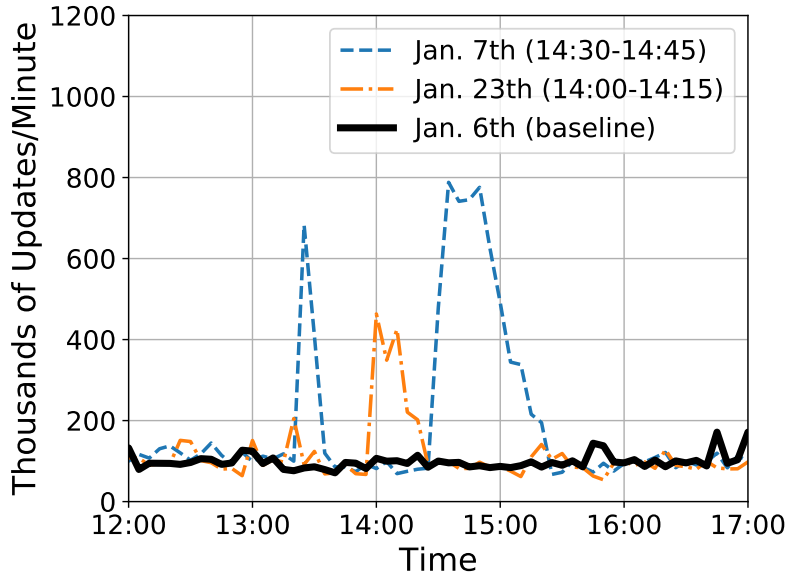
% of announcements with most-common prefix: 97% of prefixes has just one origin!

DISCO: Experimental Evaluation

- PK sent via Transitive Attribute `0xff`
  - reserved for testing and development



% of announcements with most-common prefix: 97% of prefixes has just one origin!



Prefix updates; buggy-routers caused 'peak' in both experiments (less in 2^{nd} - patching).

- Triggered bug in few FRR routers (patch exists)

Evaluation results

- Can we send *pk* in BGP announcements as transitive attribute?
  - $<<$ 1% of ASes drop announcement *or* attribute
  - Few un-patched, buggy routers failed

Evaluation results

- Can we send *pk* in BGP announcements as transitive attribute?
  - $<<$ 1% of ASes drop announcement *or* attribute
  - Few un-patched, buggy routers failed
- Can registrars certify pk from $> x$% of vantage points?
  - Used simulations of BGP topology, for reachability to 262 RouteView and RIPE RIS collectors
  - Result: Even with over 1% drop of both announcements *and* attribute, more than 95% of the vantage points report *pk*

Evaluation results

- Can we send *pk* in BGP announcements as transitive attribute?
  - $<<$ 1% of ASes drop announcement *or* attribute
  - Few un-patched, buggy routers failed
- Can registrars certify pk from $> x$% of vantage points?
  - Used simulations of BGP topology, for reachability to 262 RouteView and RIPE RIS collectors
  - Result: Even with over 1% drop of both announcements *and* attribute, more than 95% of the vantage points report *pk*
- Can attacker get DISCO-certified by prefix hijacking?
  - Prefix-hijacks: $<$ 3% certified, and 81% of these are by sole upstream provider of victim

Evaluation results

- Can we send *pk* in BGP announcements as transitive attribute?
  - $<<$ 1% of ASes drop announcement *or* attribute
  - Few un-patched, buggy routers failed
- Can registrars certify pk from $> x$% of vantage points?
  - Used simulations of BGP topology, for reachability to 262 RouteView and RIPE RIS collectors
  - Result: Even with over 1% drop of both announcements *and* attribute, more than 95% of the vantage points report *pk*
- Can attacker get DISCO-certified by prefix hijacking?
  - Prefix-hijacks: $<$ 3% certified, and 81% of these are by sole upstream provider of victim
- $\Rightarrow \mathrm{DISCO}$ appears deployable.

Conclusion

- Adoption of RPKI is critical and challenging
- Automation, validation may help adoption, reduce conflicts
- DISCOmay help: automation, validation, avoid dependency on records
  - At costs... e.g., prefix-squatters
  - Maybe adoption will improve anyway? there is hope!
  - Improving security benefits and incentives may help, too

Further work

- Specifications
- Production-ready implementation

Thank you!

Questions?

Amir.Herzberg@UConn.edu