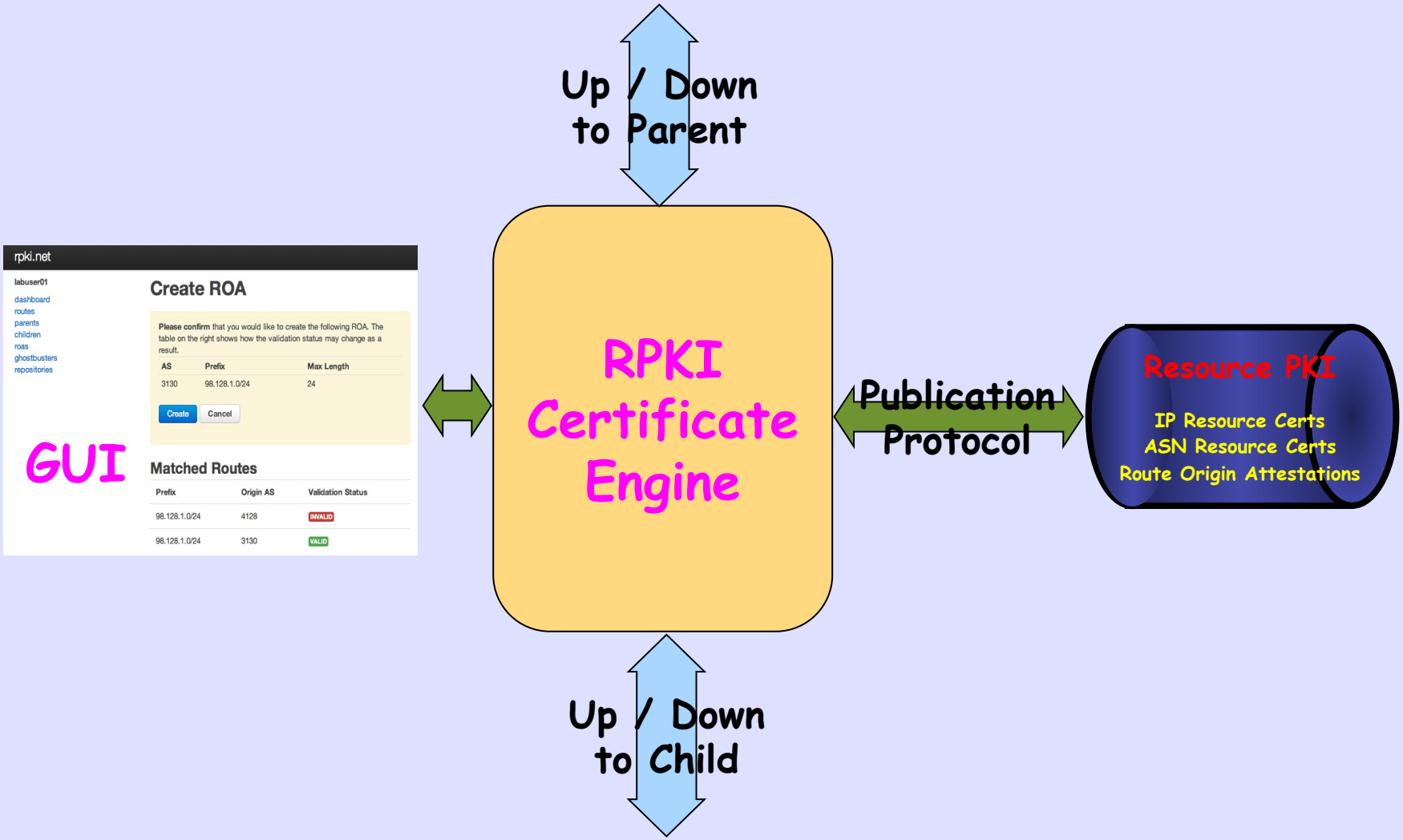
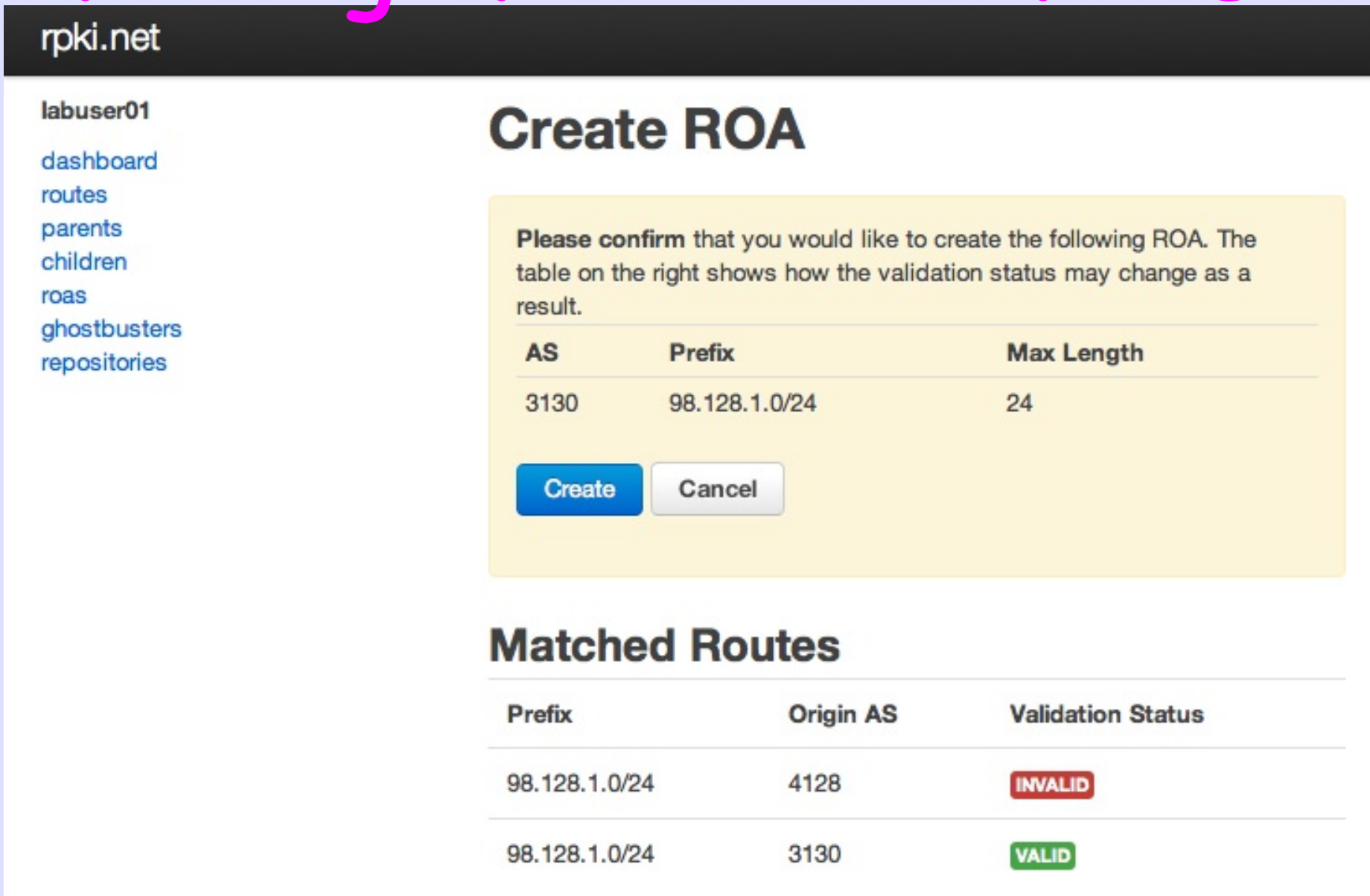
Relying / Issuing Parties and ROA Validation

2012.04.17

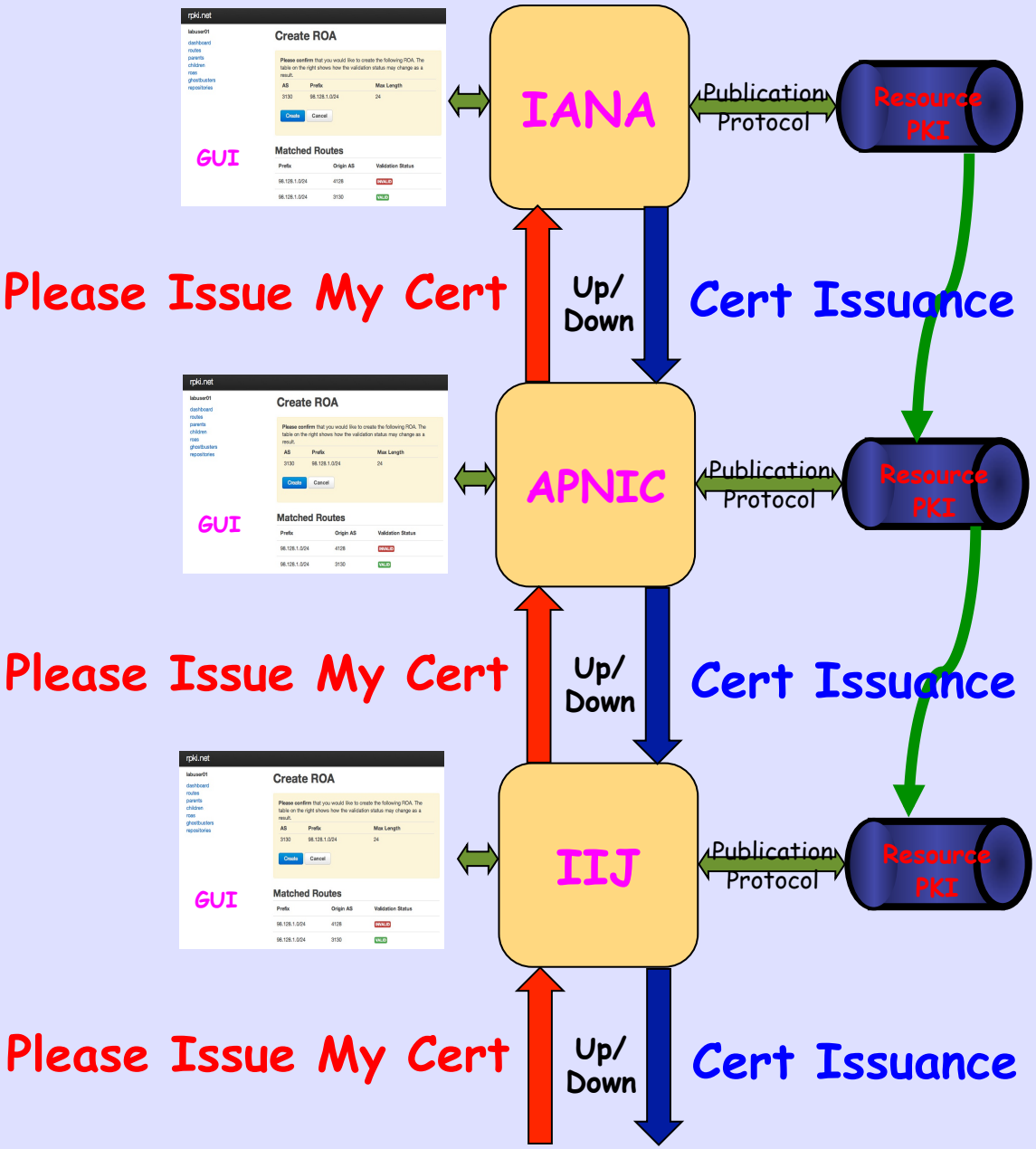
Randy Bush <randy@psg.com>


rpki.net
labuser01
dashboard
routes
parents
children
roas
ghostbusters
repositories
Create ROA
Please confirm that you would like to create the following ROA. The table on the right shows how the validation status may change as a result.
AS
Prefix
Max Length
3130
98.128.1.0/24
24
Create
Cancel
Matched Routes
Prefix
Origin AS
Validation Status
98.128.1.0/24
4128
INVALID
98.128.1.0/24
3130
VALID
GUI
Up / Down to Parent
RPKI Certificate Engine
Publication Protocol
Resource PKI
IP Resource Certs
ASN Resource Certs
Route Origin Attestations
Up / Down to Child

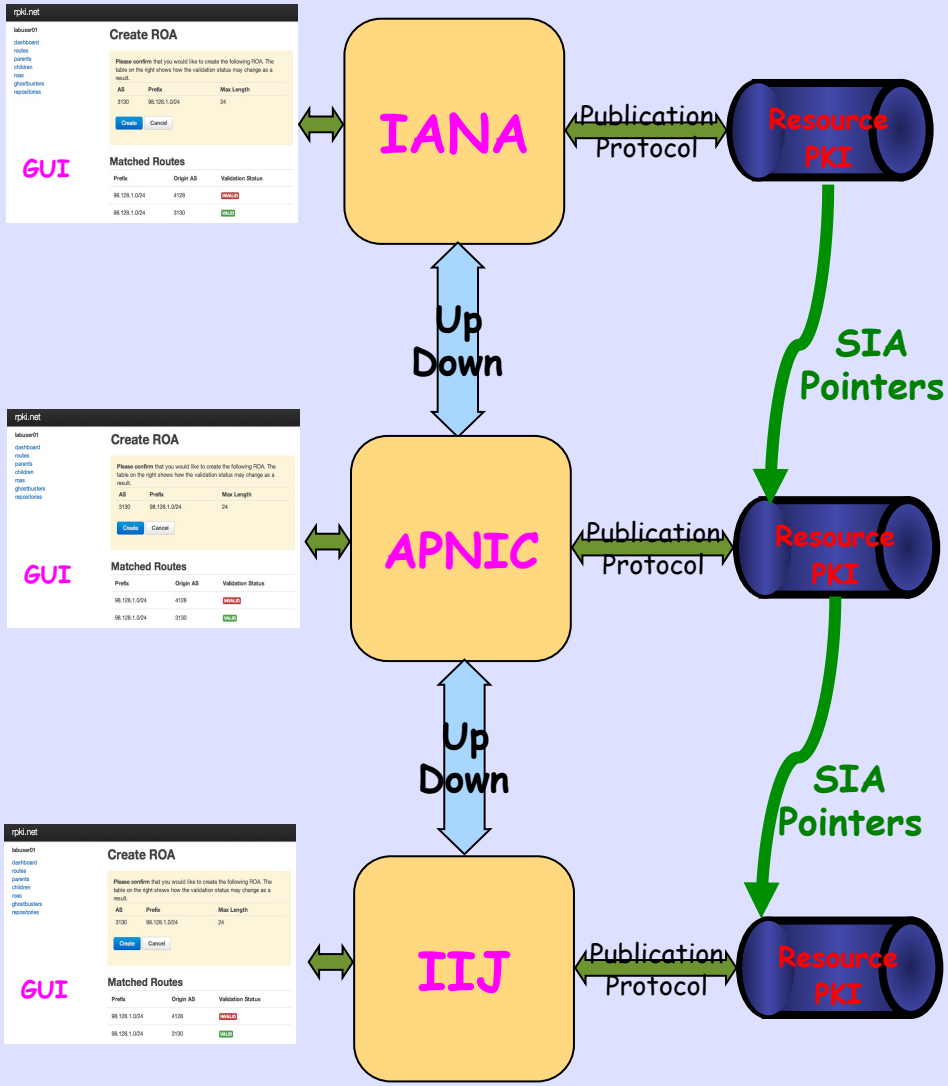
Warning What ROA Will Do



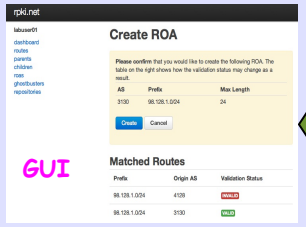
Issuing Parties



Issuing Parties

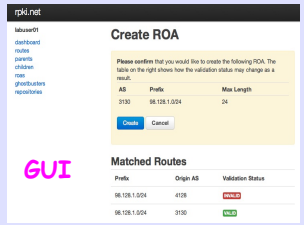


Issuing Parties Relying Parties



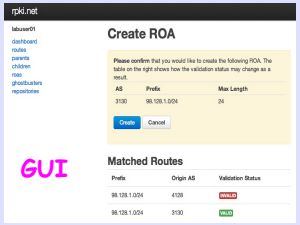
IANA

Up
Down



APNIC

Up
Down



IIJ

Publication Protocol

Publication Protocol

Publication Protocol

Trust Anchor

Resource PKI

Resource PKI

Resource PKI

SIA Pointers

SIA Pointers

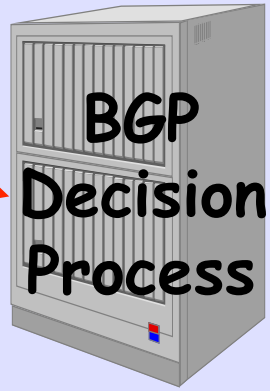
RCynic Gatherer

Validated Cache

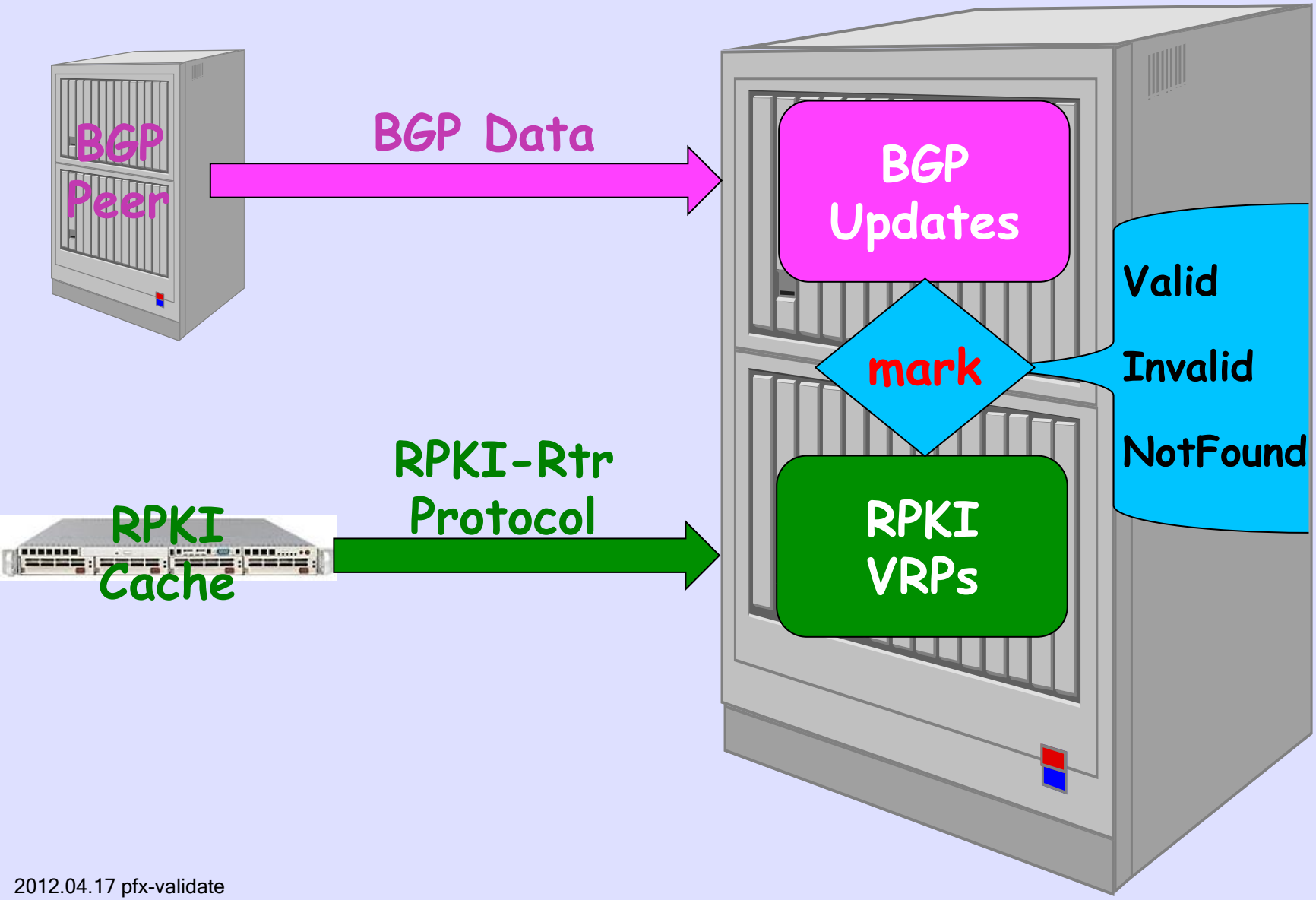
Pseudo IRR

```
route:    147.28.0.0/16
descr:    147.28.0.0/16-16
origin:   AS3130
notify:   irr-hack@rpki.net
mnt-by:   MAINT-RPKI
changed:  irr-hack@rpki.net 20110606
source:   RPKI
```

NOC Tools



Marking BGP Updates



Result of Check

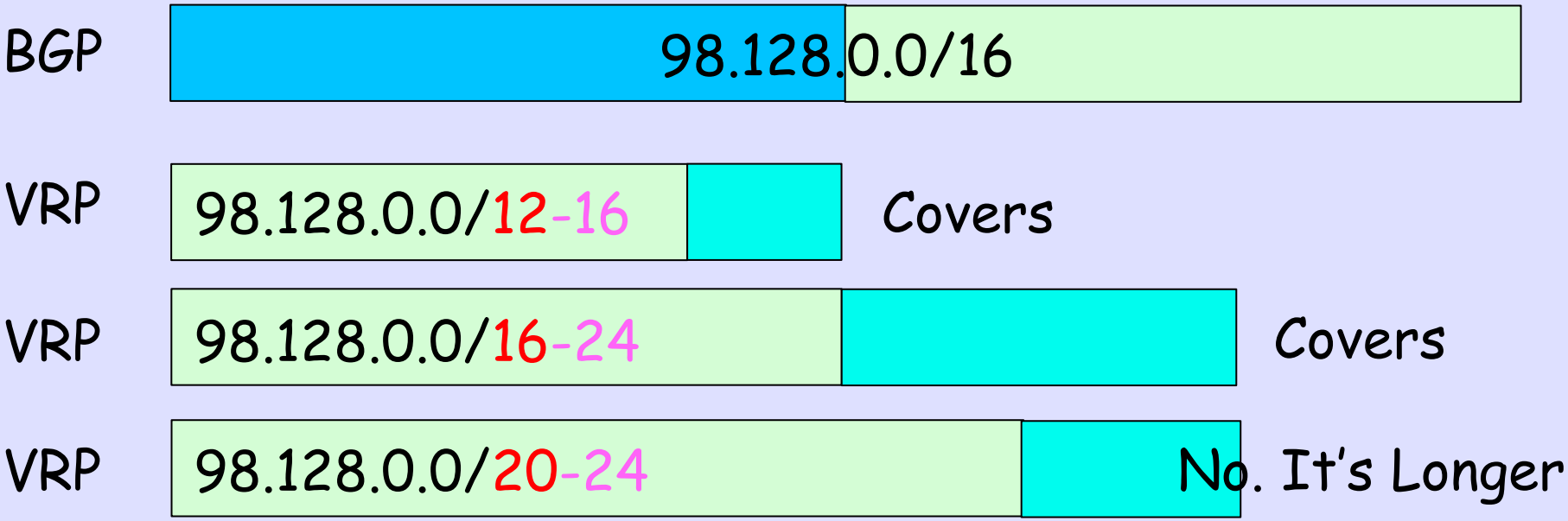
- **Valid** – A matching/covering VRP was found with a matching AS number
- **Invalid** – A covering VRP was found, but the AS number did not match, and there was no other matching one
- **NotFound** – No matching or covering VRP was found, same as today

The Operator Tests the Marks and then Applies Local Policy

What are the BGP / VRP[1] Matching Rules?

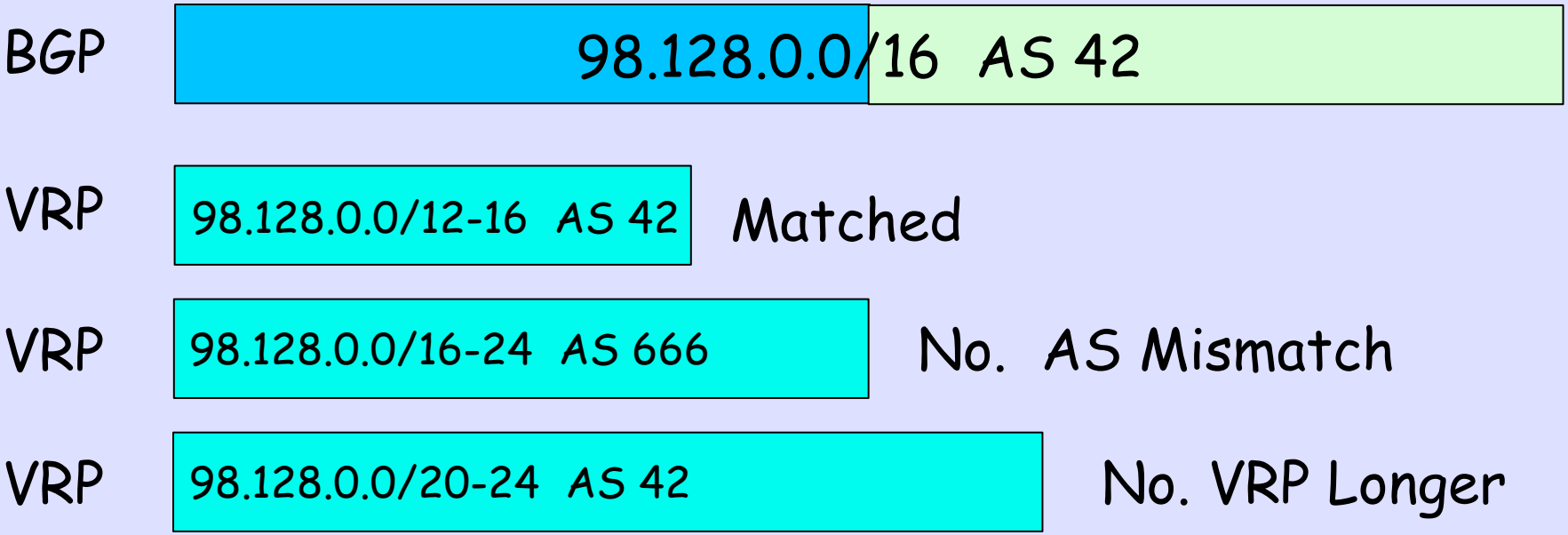
[1] Validated ROA Payload

- A Route is **Covered** by a VRP when the VRP prefix length is less than or equal to the Route prefix length
- Note: Covered does not use max-len



A Route is **Matched** by a VRP when

- the Route is Covered by that VRP,
- the Route's length is less than or equal to the VRP max-len, and
- the Route's Origin AS is equal to the VRP's AS



More Formally

```
ROA = (Rp, Rl, Ra)            // prefix, length, AS
VRPs = {Vp, Vl, Vm, Va}       // prefix, len, max-len, AS

cover(V,R) = intersect (Vp, Rp) and Vl <= Rl

match(V,R) = cover(V,R) and Rl <= Vm and Ra = Va
```

More Formality

	Rl <= Vm		Rl > Vm	
	Ra=Va	Ra~=Va	Ra=Va	Ra~=Va
cover(V,R)	Valid	Invalid	Invalid	Invalid
~ cover(V,R)	NotFound	NotFound	NotFound	NotFound

And if You Liked That

TE :== There Exists
FA :== For All

valid (R) :== TE V in VPRs such that Tag(V,R) = V

invalid (R) :== ~valid(R) and TE V in VPRs such that Tag(V,R) = Invalid

NotFound(R) :== ~valid(R) and ~invalid(R)

expanded:

valid(R) ==> TE V in VRPs such that intersect (Vp,Rp) and Vl <= Rl and
Rl <= Vm and Ra=Va

invalid(R) ==> ~valid(R) and
TE V in VRPs such that intersect(Vp,Rp) and Vl <= Rl and
(Rl > Vm or Ra ~= Va)

notfound(R) ==> FA V in VRPs, ~intersect(Vp,Rp) or Vl > Rl

Had Enough?

Matching and Validity

VRP_0	98.128.0.0/16-24 AS 6
VRP_1	98.128.0.0/16-20 AS 42

BGP	98.128.0.0/12	AS 42	NotFound, not covered by any VRP
BGP	98.128.0.0/16	AS 42	Valid, Matches VRP_1
BGP	98.128.0.0/20	AS 6	Valid, Matches VRP_0
BGP	98.128.0.0/22	AS 42	Invalid, length within VRP_1 but AS mismatch
BGP	98.128.0.0/24	AS 42	Invalid, longer than VRP_1 although AS matches
BGP	98.128.0.0/24	AS 6	Valid, Matches VRP_0

VRP with AS0

- It is supposed to mark a prefix as always invalid
- But what happens when there is a VRP for AS0 and another VRP which matches the announcement?
- The announcement is matched, and is therefore Valid

- Router implementations do not accept announcements with AS0.
- So, you will mark as Invalid when a VRP with AS0 covers as long as there is no matching VRP.
- But think of the case where a court order causes RIPE to issue a VRP with AS0 for you, but a 'rescue' trust anchor published a matching VRP. You are saved!

Don't Accept Invalid

- If your policy accepts Invalid,
- A more specific prefix hijack will be marked as Invalid
- But it will still be accepted
- Because it is the only candidate for the more specific prefix

- So maybe you don't want to accept Invalids?

Just Closed Issue(s)

- Should updates learned via iBGP be marked?
- Should updates injected into BGP on this router be marked?
- My bottom line:
  - Yes, to support incremental deployment
  - I do not want to find out I am announcing garbage when my neighbor's NOC calls

Allowing Holes

- Big Provider announces 10.0.0.0/8
- Wants to issue ROA for 10/8 before ensuring ROAs are issued for customers
- So signal hole-punching is allowed by max-len==0

10.0.0.0/8-0 42
10.0.0.0/9-0 42
10.128.0.0/9-0 42

would cause the marking of the following as Valid

10.0.0.0/8 42
10.0.0.0/9 42
10.128.0.0/9 42

and the following as NotFound

10.42.0.0/24 42
10.42.0.0/16 666
10.77.0.0/24 666

but would cause the marking of the following as Invalid

10.0.0.0/8 666
10.0.0.0/9 666
10.128.0.0/9 666

Pfui!

- This protects 10/8 but nothing else, pretty useless
- Generate temporary customer ROAs from BGP table, get real protection