

# ***RPKI: Introducción y instalación de un validador***

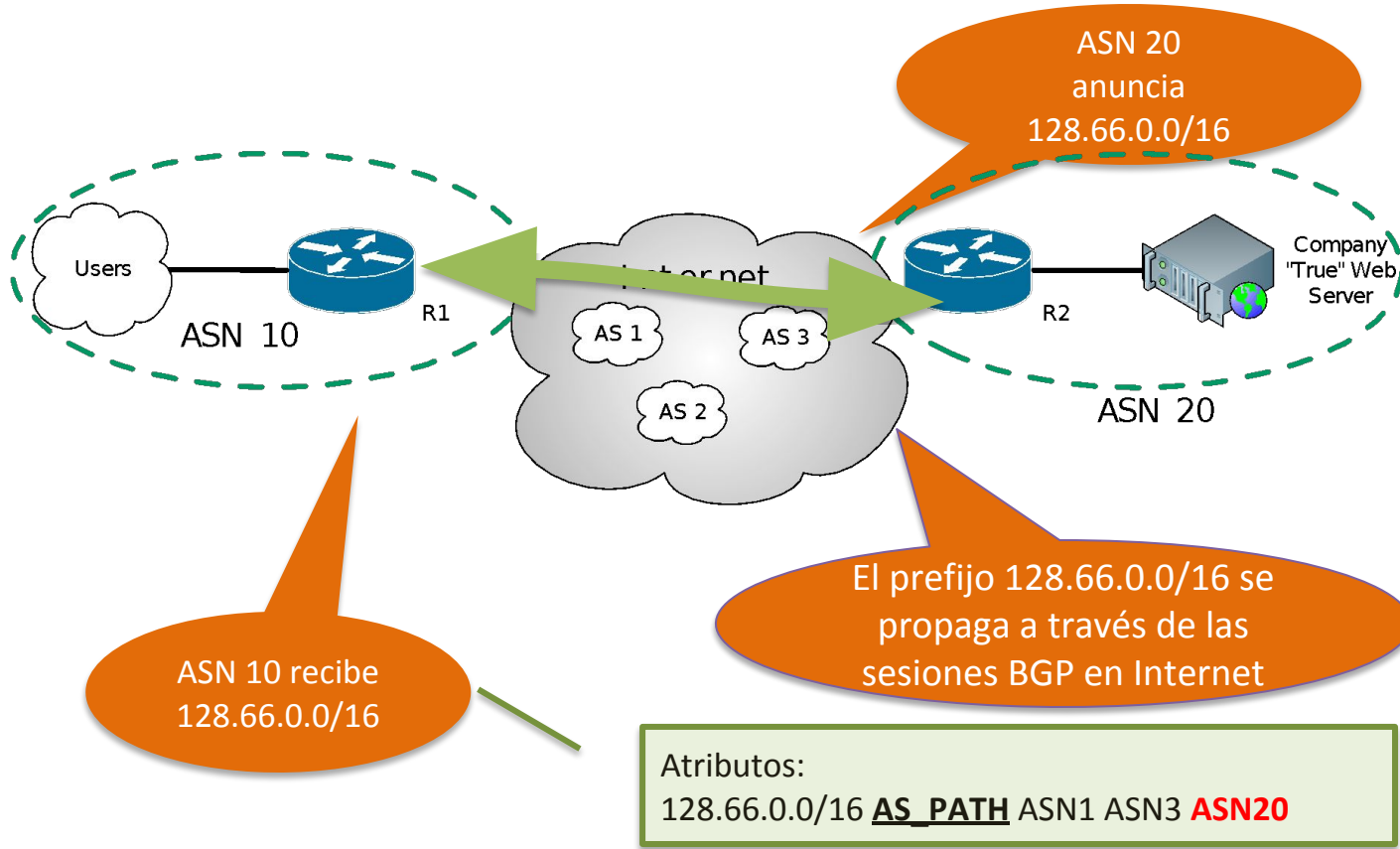
***Carlos M. Martinez***

***Webinar LACNOG***

***@CagnazzoEng***

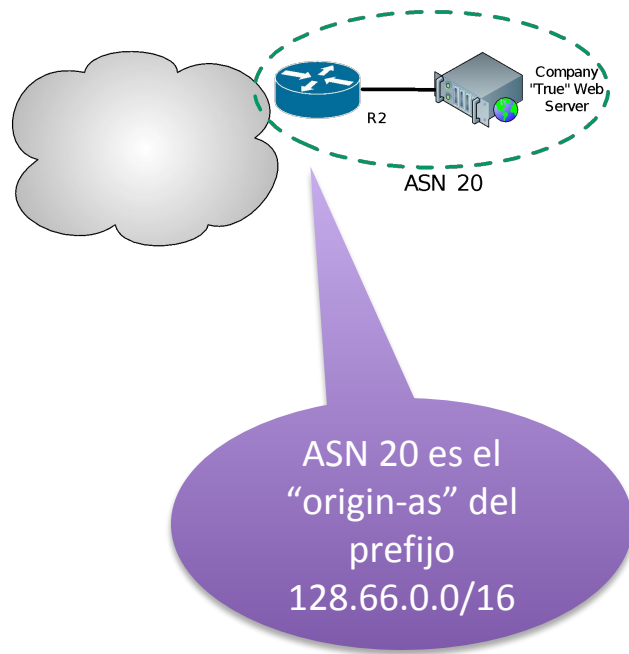


# BGP: *Routing by Rumor*

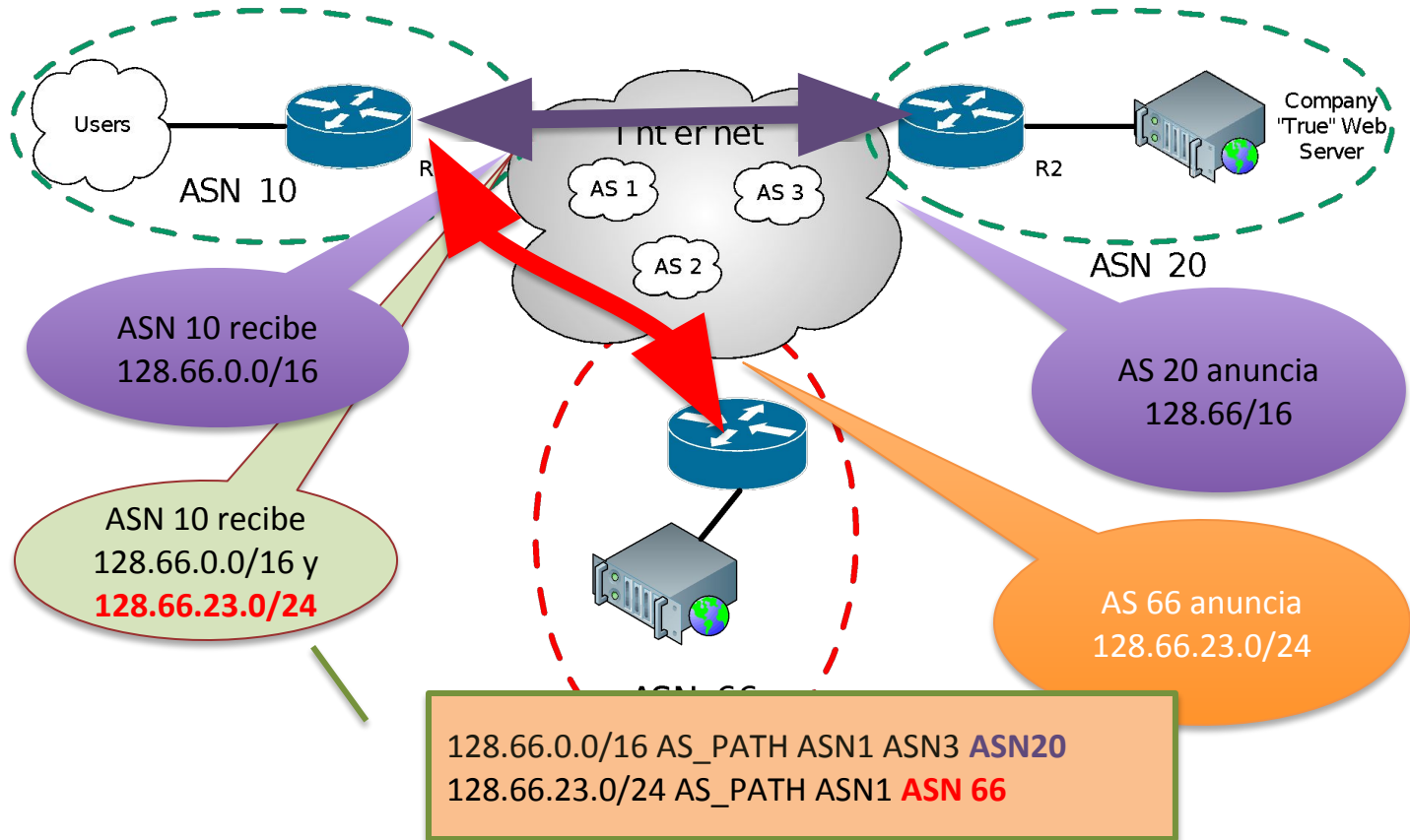


# BGP: *Routing by rumor*

- BGP elige rutas de acuerdo a su **algoritmo de decisión** y a los valores de los **atributos**
- AS\_PATH y AS de origen
  - AS\_PATH es la lista de sistemas autónomos recorridos por un UPDATE dado
  - Incluye el AS que origina el anuncio (“**origin-as**”)

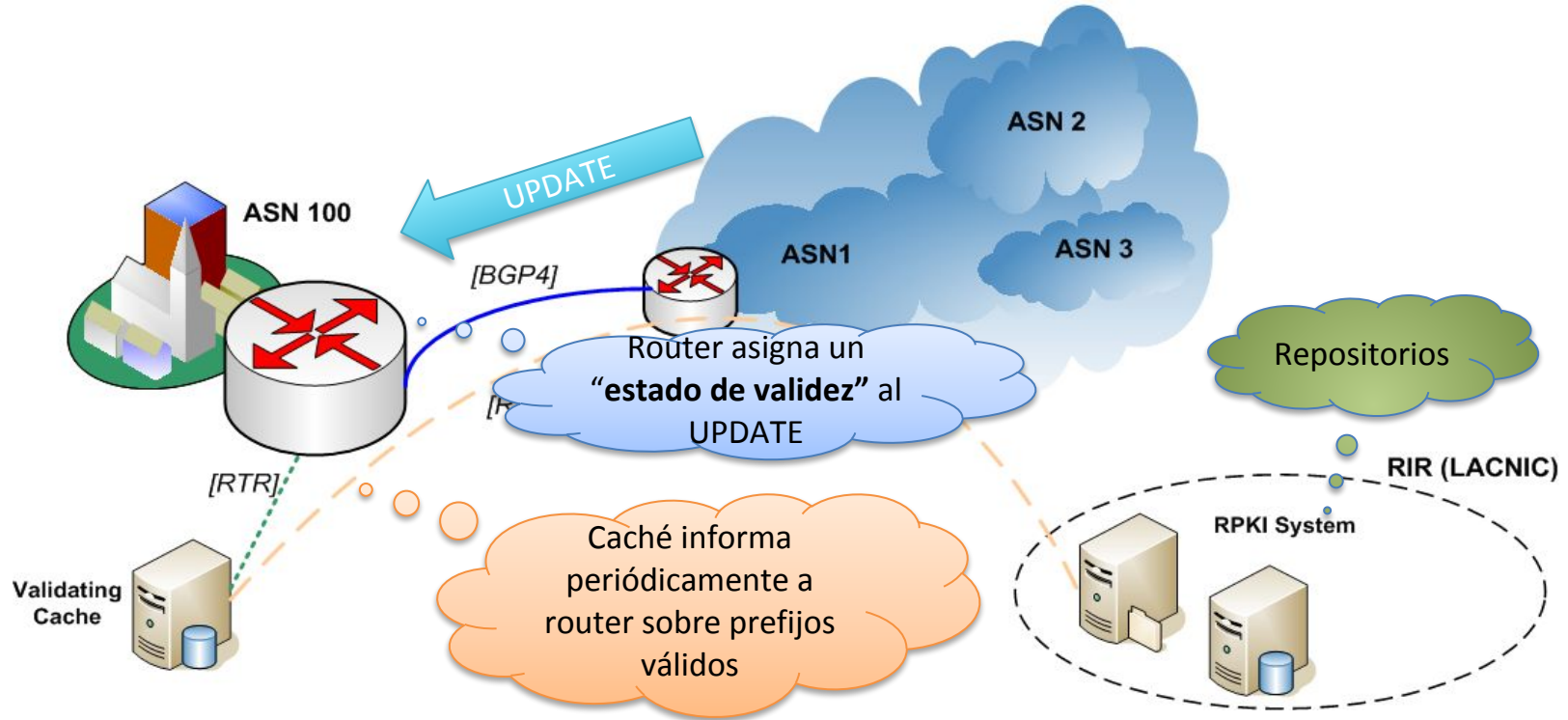


# Secuestro de rutas

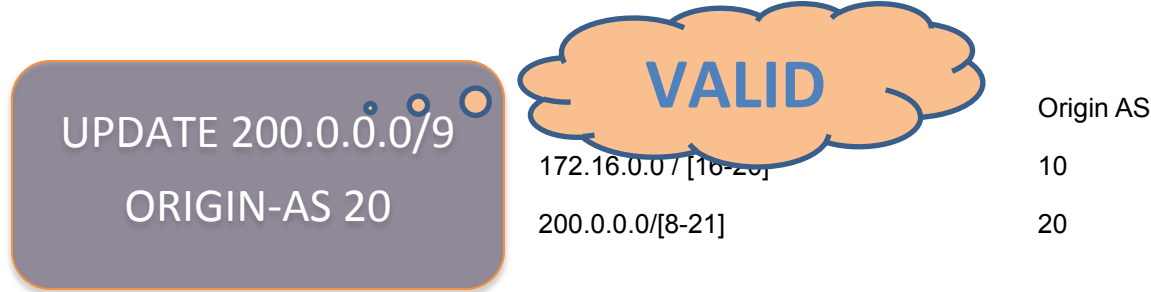


**¿QUÉ PODEMOS HACER AL RESPECTO?**

# Validación de origen (ROV)

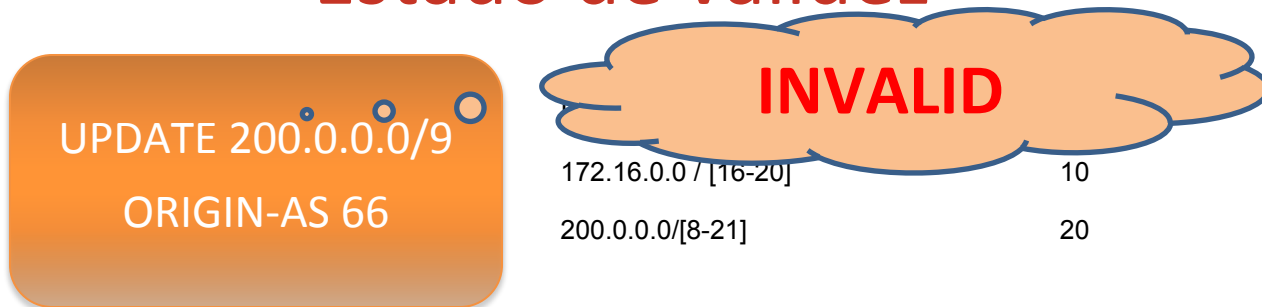


# Estado de validez



- Si el “UPDATE pfx” **no** encuentra ninguna entrada que lo cubra en la BdeD -> “**not found**”
- Si el “UPDATE pfx” si encuentra al menos una entrada que lo cubra en la BdeD y además el AS de origen del “UPDATE pfx” coincide con uno de ellos -> “**valid**”
- En el caso anterior, si **no** coincide ningun AS de origen -> “**invalid**”

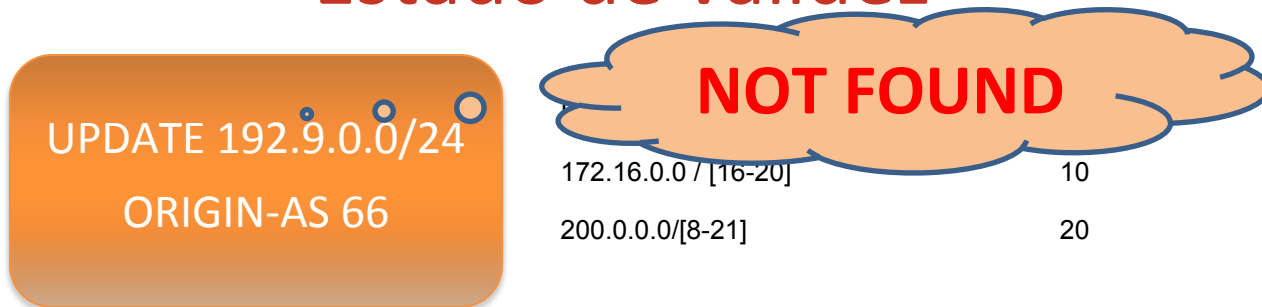
# Estado de validez



- Si el “UPDATE pfx” **no** encuentra ninguna entrada que lo cubra en la BdeD -> “**not found**”
- Si el “UPDATE pfx” si encuentra al menos una entrada que lo cubra en la BdeD y además el AS de origen del “UPDATE pfx” coincide con uno de ellos -> “**valid**”
- En el caso anterior, si **no** coincide ningun AS de origen -> “**invalid**”

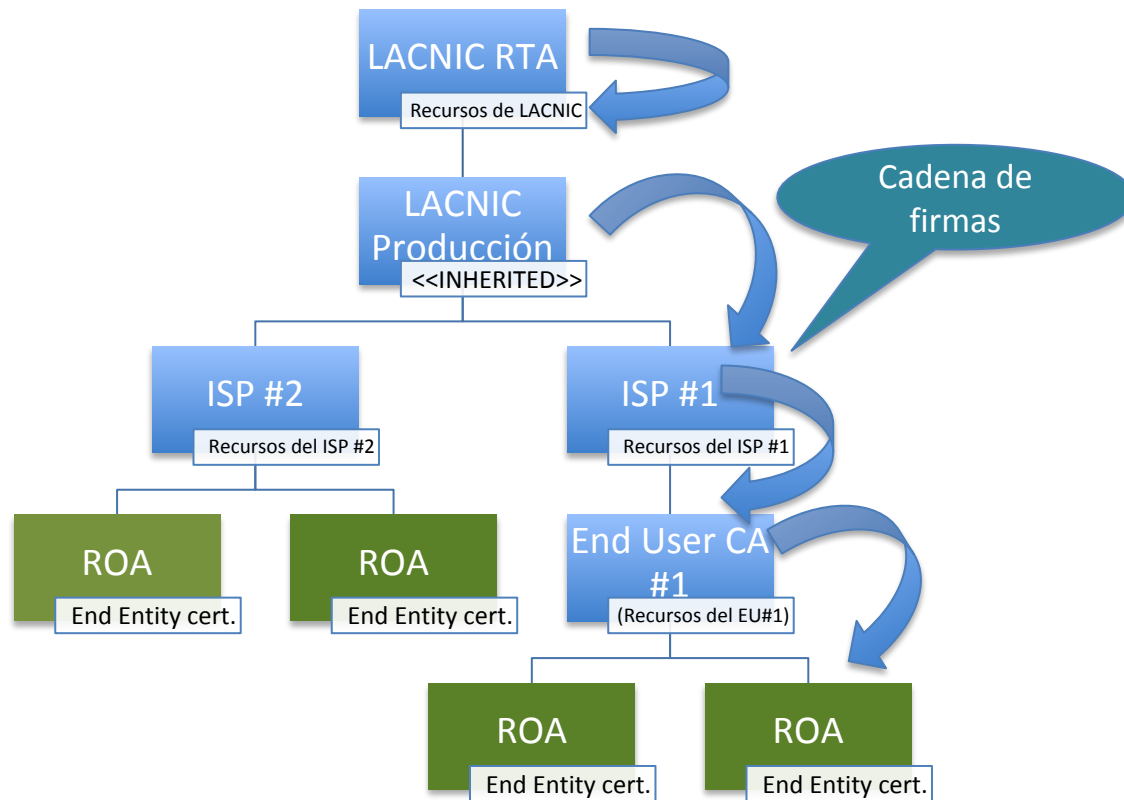


# Estado de validez



- Si el “UPDATE pfx” **no** encuentra ninguna entrada que lo cubra en la BdeD -> “**not found**”
- Si el “UPDATE pfx” si encuentra al menos una entrada que lo cubra en la BdeD y además el AS de origen del “UPDATE pfx” coincide con uno de ellos -> “**valid**”
- En el caso anterior, si **no** coincide ningun AS de origen -> “**invalid**”

# RPKI: *Estructura del repositorio*



**¿Y COMO LO HACEMOS?**

# Roles en el enrutamiento seguro en Internet

## Titular de recursos de numeración

- Ser conscientes de su política de enrutamiento
  - ¿desde que AS anuncio mis prefijos?
  - ¿como los agrego / desagrego?
- Crear los ROAs en el sitio web de LACNIC (en el caso de la región LATAM)
  - <https://milacnic.lacnic.net>
  - Los ROAs deben reflejar mi politica de enrutamiento

## Operadores de infraestructura

- Configurar ROV en los routers que hablan eBGP
  - Configurar la conexión con el validador
  -
- Instalar y configurar una o más instancias de alguna herramienta de validación
- Monitorear el funcionamiento de su infraestructura

# Herramientas de validación

- RIPE RPKI Validator
  - <https://www.ripe.net/manage-ips-and-asns/resource-management/certification/tools-and-resources>
- Validador FORT
  - <https://fortproject.net/validator>
- Routinator 3000
  - <https://www.nlnetlabs.nl/projects/rpki/routinator/>
- Otros:
  - RPKI-Client, RPSTIR, Octo-RPKI

# Routinator 3000

- Desarrollado por NLNetlabs
- Soporta validación y servidor RTR
- Se instala en sistemas Unix (MacOS incluido)
- Demo:
  - Ubuntu 18.04
  - Ansible

**ROUTINATOR**



# Instalación de Routinator

Pre-requisitos:

- Ubuntu 18.04
- compiladores
- rsync
- tmux\*
- compilador y utilidades de Rust

[Demo instalación ANSIBLE]

# Configuración de Routinator 3000

Una vez que está instalado realizamos los siguientes pasos:

Validación del repositorio:

- `routinator init [--accept-arin-rpa]`
- `routinator -v vrps`

Servidor RTR

- `routinator server --rtr 0.0.0.0:3323 --refresh=900`



# Demo: Routinator 3000

- El conocimiento para instalarlo está representado en un *playbook* de [Ansible](#)
- El demo se hace contra una VM de [Vagrant](#) usando el provisioner Ansible de Vagrant
  - Se usa una base Ubuntu 18.04
  - “vagrant up” o “vagrant up --provision”
- El mismo *playbook* se puede ejecutar por SSH contra un Ubuntu usando el siguiente comando:
  - “ansible-playbook ...”

¡Muchas gracias!

