SEND-SAVI

Guidelines for SAVI-SEND specification

- Perimetrical deployment
  - Trusted and Validating ports
  - Cryptography checks are performed only in the SEND SAVI device closer to the host
- Implicit assumption: address ownership can only be proved from one location
  - Not considering yet anycast addresses
- Bindings are created after a NUD exchange between the SEND host and the closer SEND-SAVI device
    - Protection against reply attacks
- Operational description for local traffic processing using a state machine
- “Advanced” features
  - Supports concurrent testing of different ports
  - Supports blocking of ports from which a test failed, to rate-limit attacks

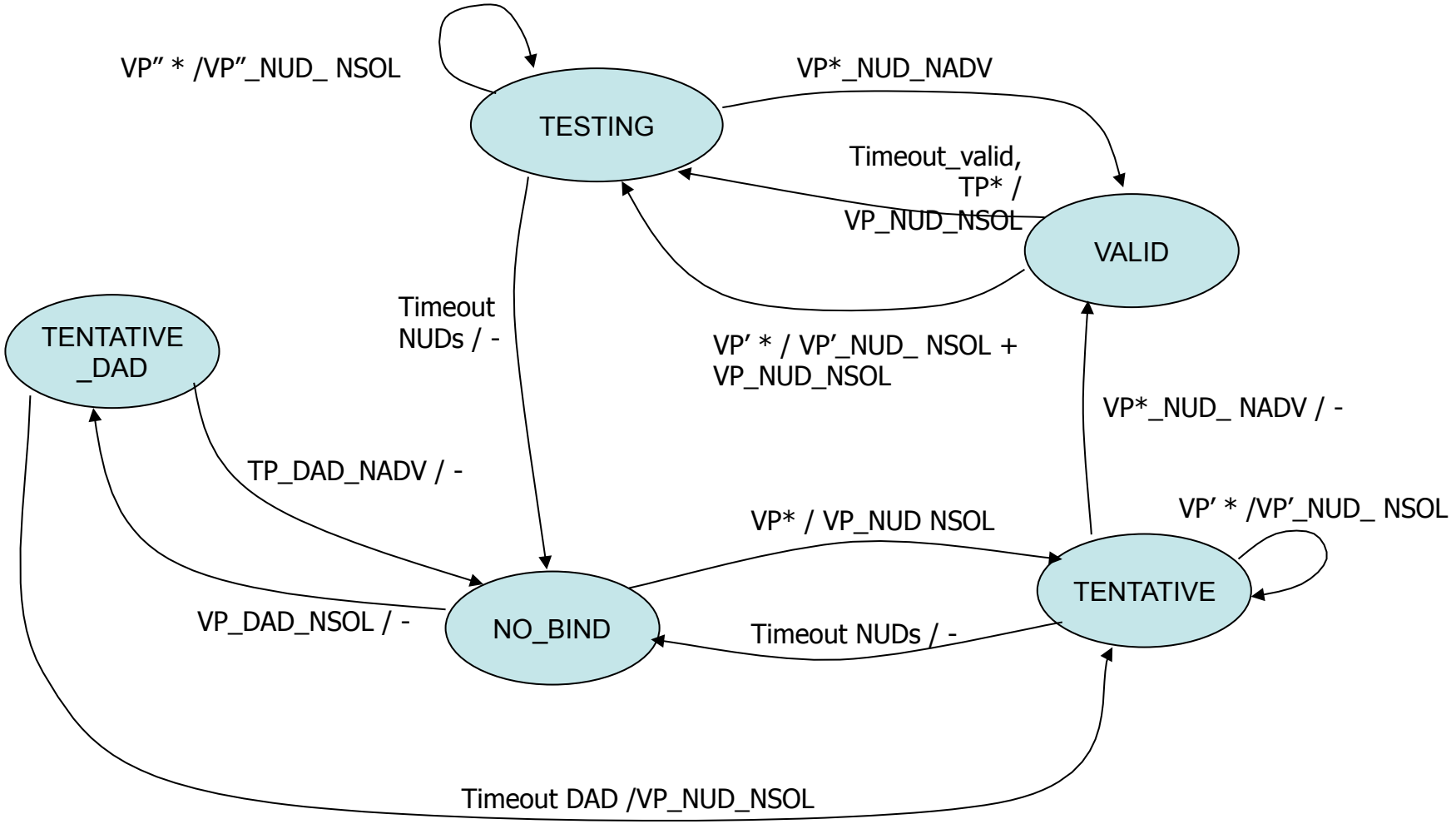
Short description of the SEND SAVI state machine for local addresses

State for local addresses

- NO_BIND: Default state
- TENTATIVE_DAD: period since a DAD NSOL is received from a Validating port until a validated DAD NADV is received, or a timer expires
- TENTATIVE: A packet (different from DAD NSOL) is received for an address for which a binding did not exist, and a NUD NSOL is issued to the port from which the packet was received.
- VALID: The binding for the source address has been verified, it is valid and usable for filtering traffic.
- TESTING: The node enters in this state when
  - a packet has been received by any port different from current port in the binding (either other Validating port or a Trusted port), or
  - the timeout for the validity of the binding has expired.

  A NUD NSOL is issued to the current port and other Validating ports from which packets with this source address have been received. Packets from the current validating ports are still forwarded.

Simplified state machine



Compatibility with SLAAC and SEND hosts

"Modes" can be defined for SAVI operation

- SLAAC-only mode
  - Binding when Timeout_DAD expires without receiving DAD NADVs (as a result of DAD NSOL)
- SEND-only mode
  - Relies on performing NUD explicitly to the host sending packets
- SLAAC/SEND-compatible mode
  - Why a different mode?
    - Because when a data packet is received from the first time, you don't know if the host is SEND or SLAAC
  - SEND hosts always have preference in bindings over non-SEND hosts
  - SLAAC competing hosts behave as in SLAAC-only mode