

## **BGP-SRx**

BGP - Secure Routing Extension

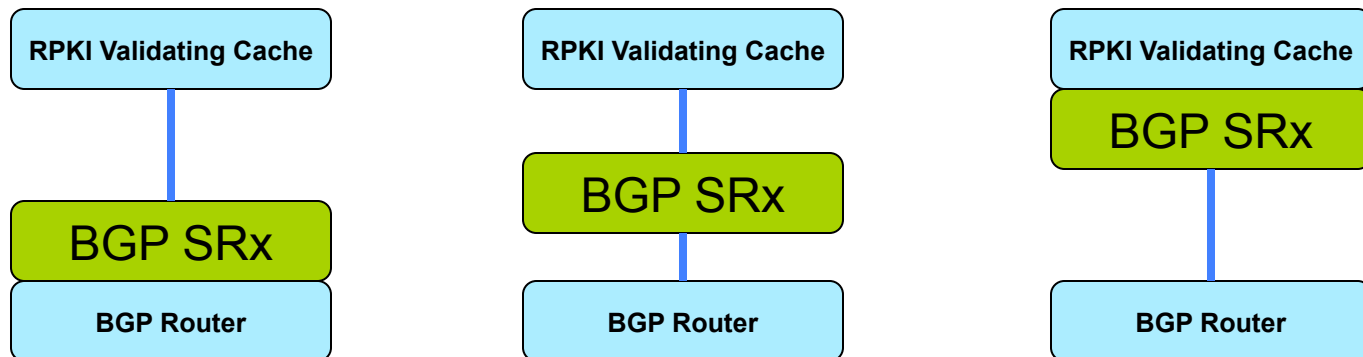
## **BRITE**

BGP Security / RPKI Interoperability Test & Evaluation

Doug Montgomery (dougmont@nist.gov)

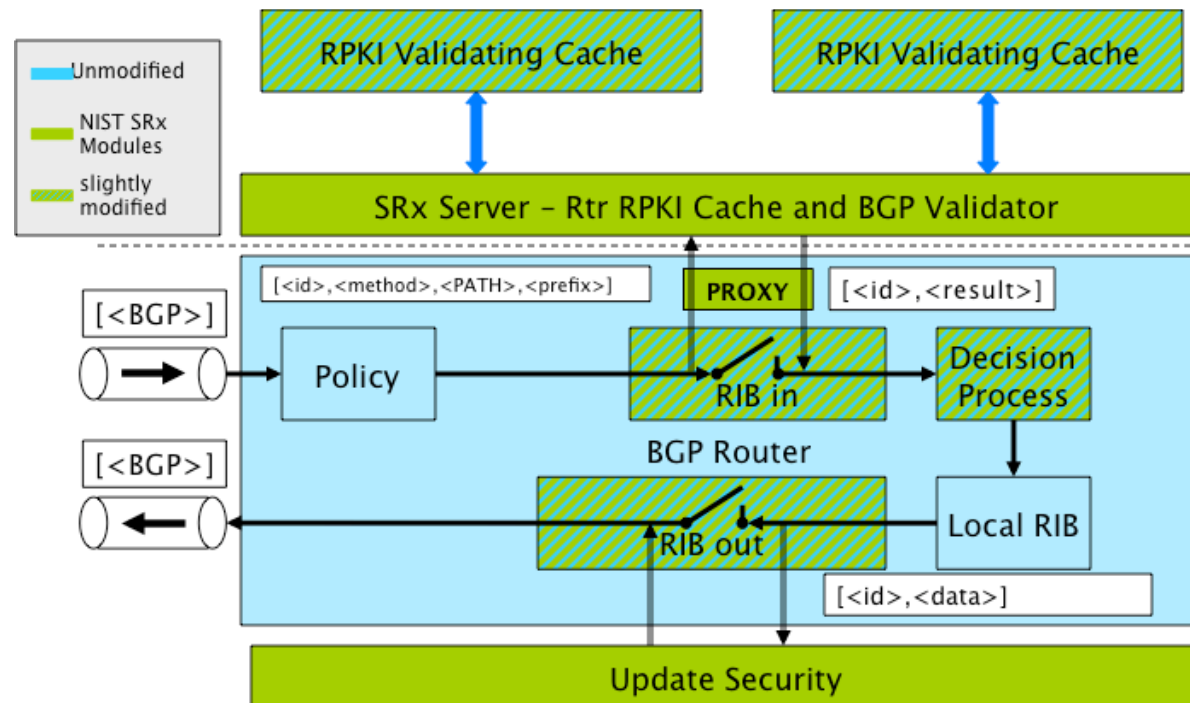
# BGP SRx Overview

- **BGP Secure Routing Extension (SRx)**
  - Software router with extensions for: RPKI Rtr cache maintenance, validation of updates, new BGP route policies.
  - SRx – implemented as extension for Quagga routing platform. Designed to support other platforms (e.g., XORP, etc).
  - Designed to support experimentation with different architectural configurations of SRx and RPKI components.
- **Status**
  - BGP SRx frame work with RPKI cache and ROA processing implemented.
    - draft-ietf-sidr-rpki-rtr-11
    - draft-ietf-sidr-roa-validation-10.txt, draft-ietf-sidr-pfx-validate-01
  - TBD
    - draft-ietf-sidr-origin-validation-signaling-00

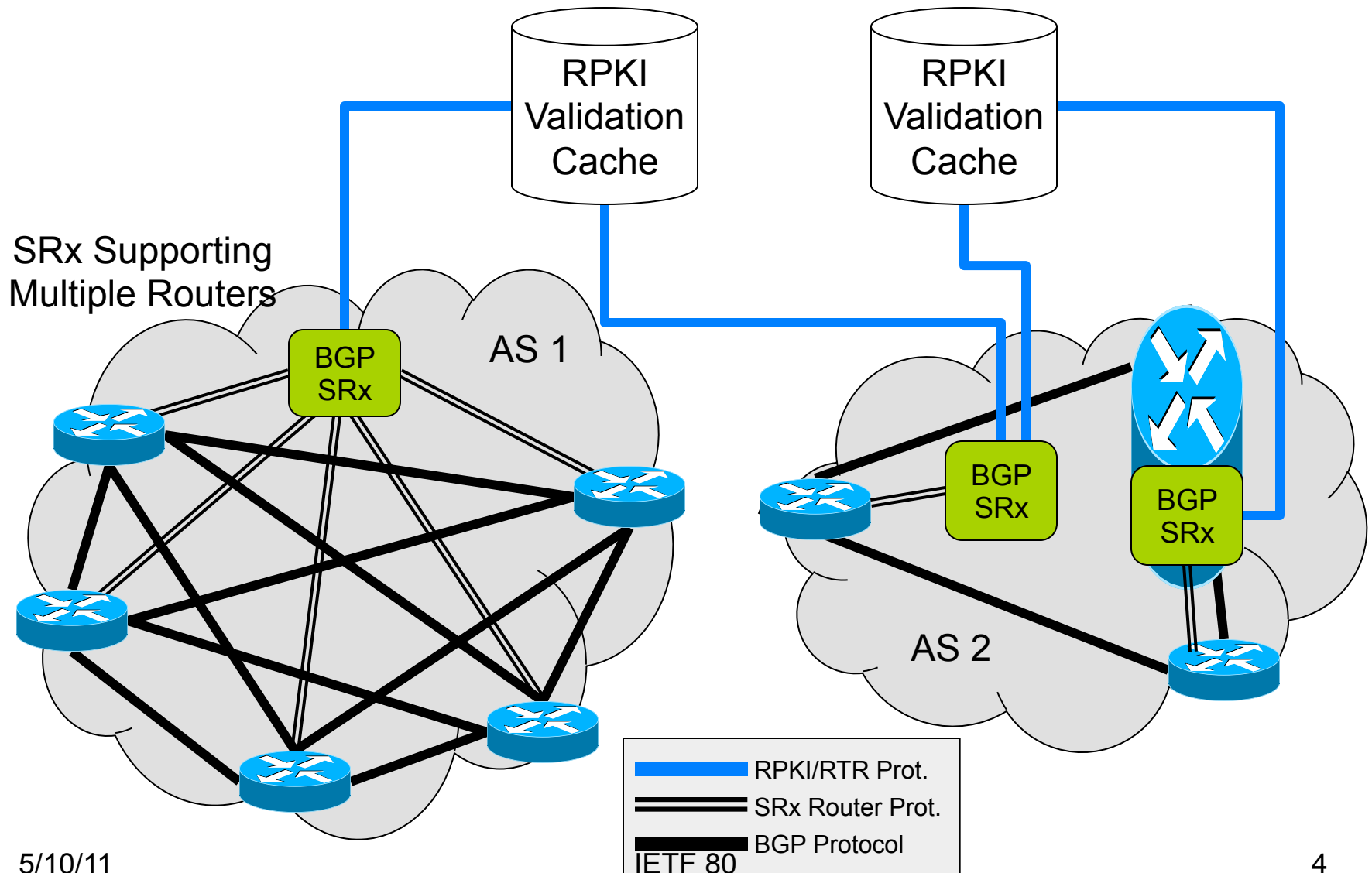


# BGP SRx Implementation

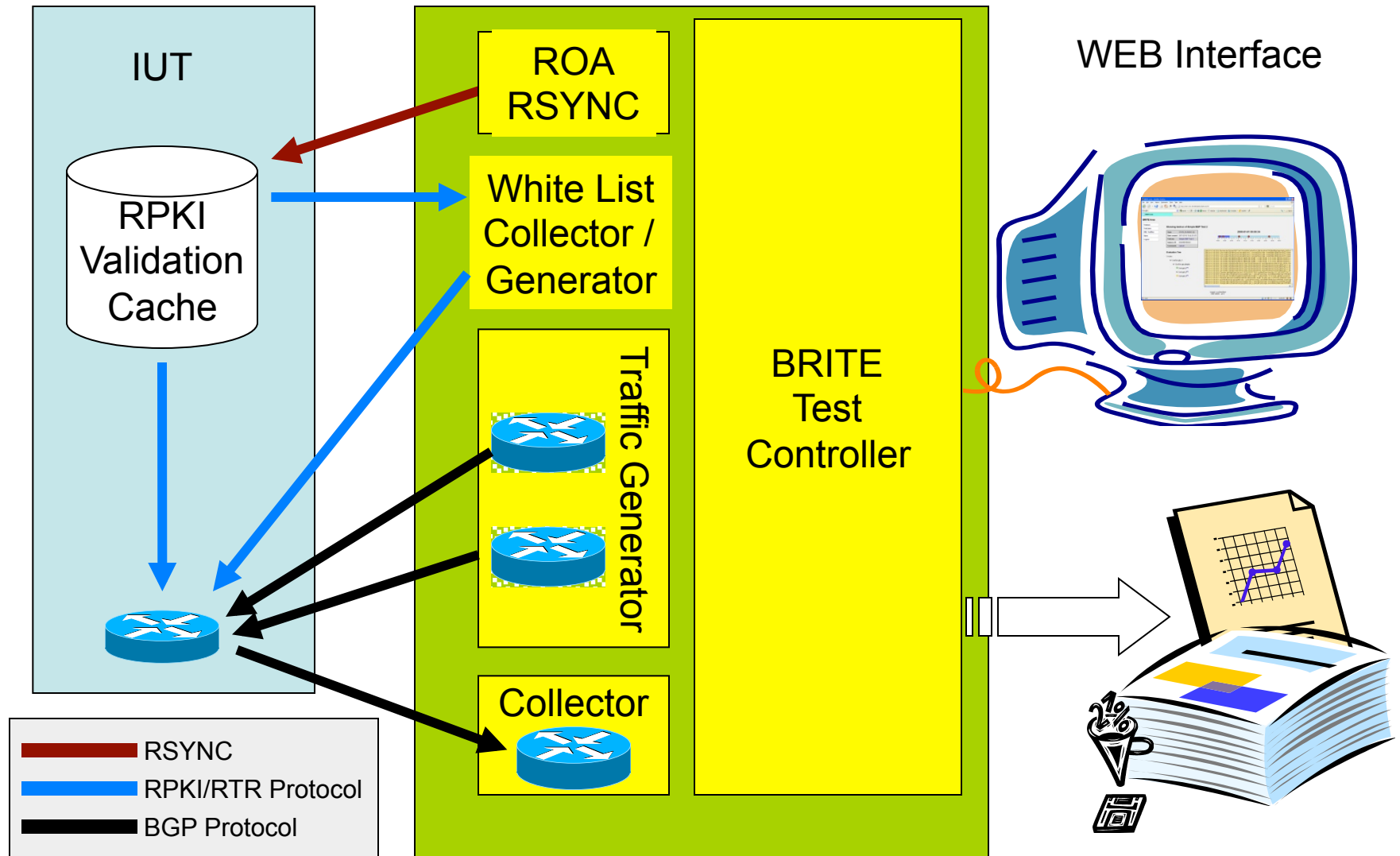
- SRx Server
  - Independent process – through proxy shim in router.
  - Supports asynchronous validation (lazy or blocking).
  - Supports multiple caches .... and multiple routers.
- Policies
  - Ignore Invalid
  - Ignore Unknown
  - Modify LocPref
  - Tie Break



# SRx Deployment Options



# BRITE Design Overview



# BRITE Overview

- **BGPSEC / RPKI Interoperability Test & Evaluation**
  - Distributed test and evaluation framework for:
    - RPKI / BGP Security implementation testing,
    - Configuration and deployment testing.
  - Flexible XML based test / scenario scripting language.
  - Can test all components / interfaces of BGP security system.
    - RPKI Validating Caches.
    - Cache to Router Protocol.
    - ROA Processing in BGP Router.
- **Online Testing Service.**
  - WWW interface to BRITE.
  - Multi-user infrastructure.
  - Real time test monitoring & reporting.
  - Other diagnostics – log files, traffic traces available for download.

# BRITE Web Interface

The screenshot displays the BRITE Web Interface in a Mozilla Firefox browser window. The interface is titled "BRITE area" and shows the details for a test run of "Simple BGP Test 2".

**Test Progress:** A progress bar at the top right indicates the test is running. The state is "STATE\_RUNNING (8)", created on "2011-03-16 15:44:25 UTC", and the instance ID is "IHSHM2V89U63".

**Test Timeline:** A horizontal timeline at the bottom shows the test's duration from 00:00 to 02:40. It features colored markers: a blue 'M' for Multiple, a green 'A' for Activation, a yellow 'B' for BGP, and a red 'W' for Whitelist.

**Events:** A legend on the right explains the markers: M=Multiple, A=Activation, B=BGP, W=Whitelist.

**Experiment Log:** A scrollable log window at the bottom right shows the test's execution details, including goal creation, job creation, and state changes.

**Goal Tree:** A tree view on the left shows the test's structure, including "GoalSet gst\_simple" and its sub-goals "Goal gol\_1", "Goal gol\_2", and "Goal gol\_3".

**Annotations:** Arrows point from the following labels to the corresponding parts of the interface:

- Goal Tree: Points to the tree view on the left.
- Wait to be activated: Points to the "Goal gol\_1" node.
- Currently processing: Points to the "Goal gol\_2" node.
- Finished successful: Points to the "Goal gol\_3" node.
- Test Progress: Points to the progress bar.
- Test Timeline: Points to the horizontal timeline.
- Events: Points to the legend.
- Experiment Log: Points to the log window.

Goal Tree

Wait to be activated

Currently processing

Finished successful

Test Progress

Test Timeline

Events:  
M=Multiple  
A=Activation  
B=BGP  
W=Whitelist

Experiment Log