Federation of Emulabs
and
Relevant New Development

Jay Lepreau
with Rob Ricci, Mike Hibler, Leigh Stoller
University of Utah

USC/ISI Federation Workshop
December 11, 2006

Emulab Federation Design "Levels"

- Level 1 - quick hack
- Level 2 - good design and function
- Level 3 - Do Everything Right and be GENI-compatible

Our Design's Goals

- Level 2 for Emulabs, including DETER
- Work pretty well for federation with PlanetLab (for which we're funded)
- Be on path to GENI compatibility

Rob will describe in next talk

Why Federate?

- Obvious: Resources, resources
  - Larger common pool
  - Better statistical multiplexing
  - Access to different (heterogenous) resources
    - Includes validation activity
  - Larger expts possible

Why Federate (less obvious)

- Access to Emulab system features not available locally
  - Out of date
  - Alpha/beta test features
  - Buggy (due to old code or new code)
  - Against policy
  - Different feature sets (beware the fork!)
- Ease testing for site-specific behavior (bug, ….)
- One mechanism eliminates version skew!
- Help build community mindset
- Partial/possible prototype for GENI federation

Why *Not* Federate?

- Stay separate (option 1)
  - No hard or ambiguous policy problems, including resource policies
  - No problems of version skew
  - Better privacy, esp. vs. testbed opers
  - Keep local users ignorant of possible better options
  - Simpler for the software
- Just merge
  - Physically (option 2a)
    - For political and economic reasons, distributed resources will always exist
    - Still, some testbeds could merge
  - Logically (option 2b)
    - See later under “ASP model”

Approaches / User Interfaces

- Single portal for multi-Emulab expts
- Single master Emulab and all others are proxies
- ASP model (variant of above)
- Peers: submit anywhere have privileges
- Many masters: submit only from “home” Emulab

Requirements

- To be incentive-compatible,
  - Local site's users' must not get any worse access to resources than they would if non-federated
  - Other risks must be mitigated

Threats

- Security
  - boss.emulab.net (only marginally higher threat from alien users)
  - ops.emulab.net (don't share)
  - fs.emulab.net (don't share)
- Alien operators
- Public Internet
- Poorly-run Emulabs
  - Security, fidelity

Risks

- API version skew
  - Mitigate with external API only, not DB state
  - Mitigate with Elab-in-Elab testing
- Confusing to user
  - Policies, mechanisms, portals
- Software complexity
- Operational complexity
  - Eg, error reporting

Federation-Relevant New Emulab Development

New: admin

- Licensing: open source
  - release by January
  - Probably Affero GPL or similar
  - Daily (or live) update of CVS repo
- Note implications for security
  - …
  - White box testing required

Recent development (low tech)

- Move to uuid for users, projs, groups
  - For federation, expt archive
  - Email names for users
- Refactoring all the code into classes and instances

Security validation of the Emulab web site [1]

- Problem: Block SQL injection attacks
  - Web page input fields -> PHP -> MySQL queries
  - Unchecked inputs allow hijacking the DB.
- Solution: Full input field checking
  - Almost all fields are checked in the PHP code.
  - Show that *all* input fields are checked.
  - Automate the checking to maintain the assertion.

  - About 70% (?) complete

Security validation of the Emulab web site [2]

- Our approach: automated black-box/white-box scanning.
  - Probe a captive Emulab-in-Emulab web site and DB.
- Black-box:
  - Spider HTML pages; find forms and input fields.
  - Use an attack web-proxy to capture hidden GET/POST fields.
- White-box:
  - Scan the sources for forms to ensure complete coverage.
  - Accumulate a dictionary of valid input field values.
- Automation:
  - Script: activation, spidering, coverage checking, and probing.
  - Probes mix in one penetration string with other valid inputs.
  - Catch unchecked probes in DB Query common code.

More and Better Hetero Resources

- Fed with PlanetLab: both directions
- Imminent wireless testbed expansion (80-120 nodes)
  - 802.11
  - SDR

New (hi tech)

- Stateful swapout / pre-emption
  - Local disk state, memory and processor state, consistent network state, time adapter/transducers
  - time travel coming…
    - Branching LVM
- Experimentation Workbench [TR Dec'06, Usenix'06]
  - Total record/replay; workflow
    - Enables assured pipelines, validation, stamp-of-approval
  - Possible staging/tracking of persistent file access
- Flexlab [HotNets'06]
  - Decouple network model from Emulab
  - Real Internet conditions and traffic from/on PlanetLab

Starting, slowly…

- Layer 2 and layer 3 devices first class Emulab objects
- Use it to configure / assure / audit Emulab infra itself