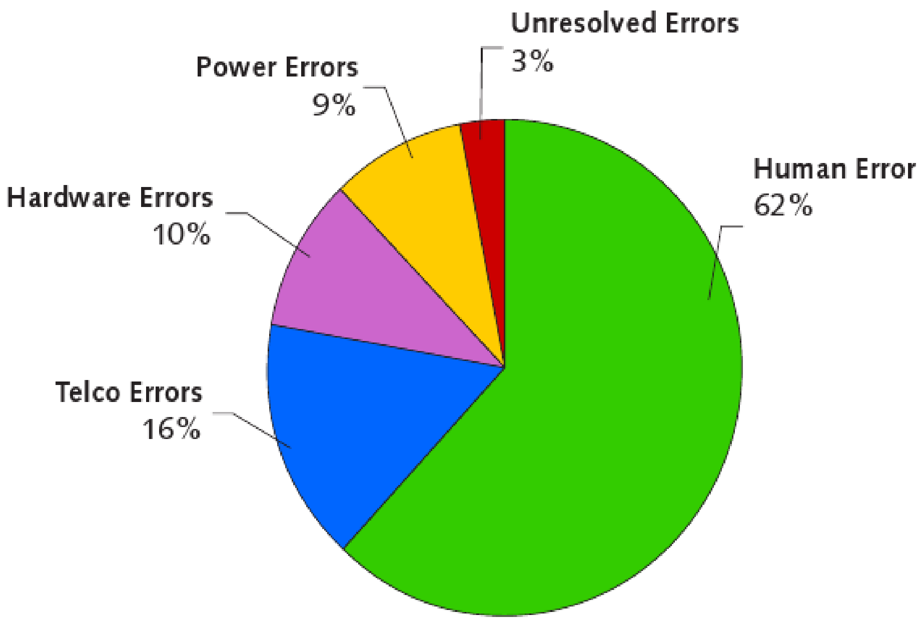
GENI Security Configuration In a Box

Ehab Al-Shaer

Assurable Networking Recent Center (ARC)
School of Computing, DePaul University,
Chicago, IL, USA

NSF GENI Workshop
University of California Davis,
January 22, 2009

State of Security Configuration Management



"Eighty percent of IT budgets is used to maintain the status quo.",
Kerravala, Zeus. "As the Value of Enterprise Networks Escalates, So Does the Need for Configuration Management." ***The Yankee Group*** **January 2004 [2].**

"Most of network outages are caused by operators errors rather than equipment failure.",
Z. Kerravala. Configuration Management Delivers Business Resiliency. The Yankee Group, November 2002.

- "It is estimated that configuration errors enable 65% of cyber attacks and cause 62% of infrastructure downtime", Network World, July 2006.
- *Recent surveys show Configuration errors are a large portion of operator errors which are in turn the largest contributor to failures and repair time [1].*
- *"Management of **ACLs** was the most critical missing or limited feature, Arbor Networks' Worldwide Infrastructure Security Report, Sept 2007.*

[1] D. Oppenheimer, A. Ganapathi, and D. A. Patterson. Why Internet services fail and what can be done about these? In *USENIX USITS*, Oct. 2003.

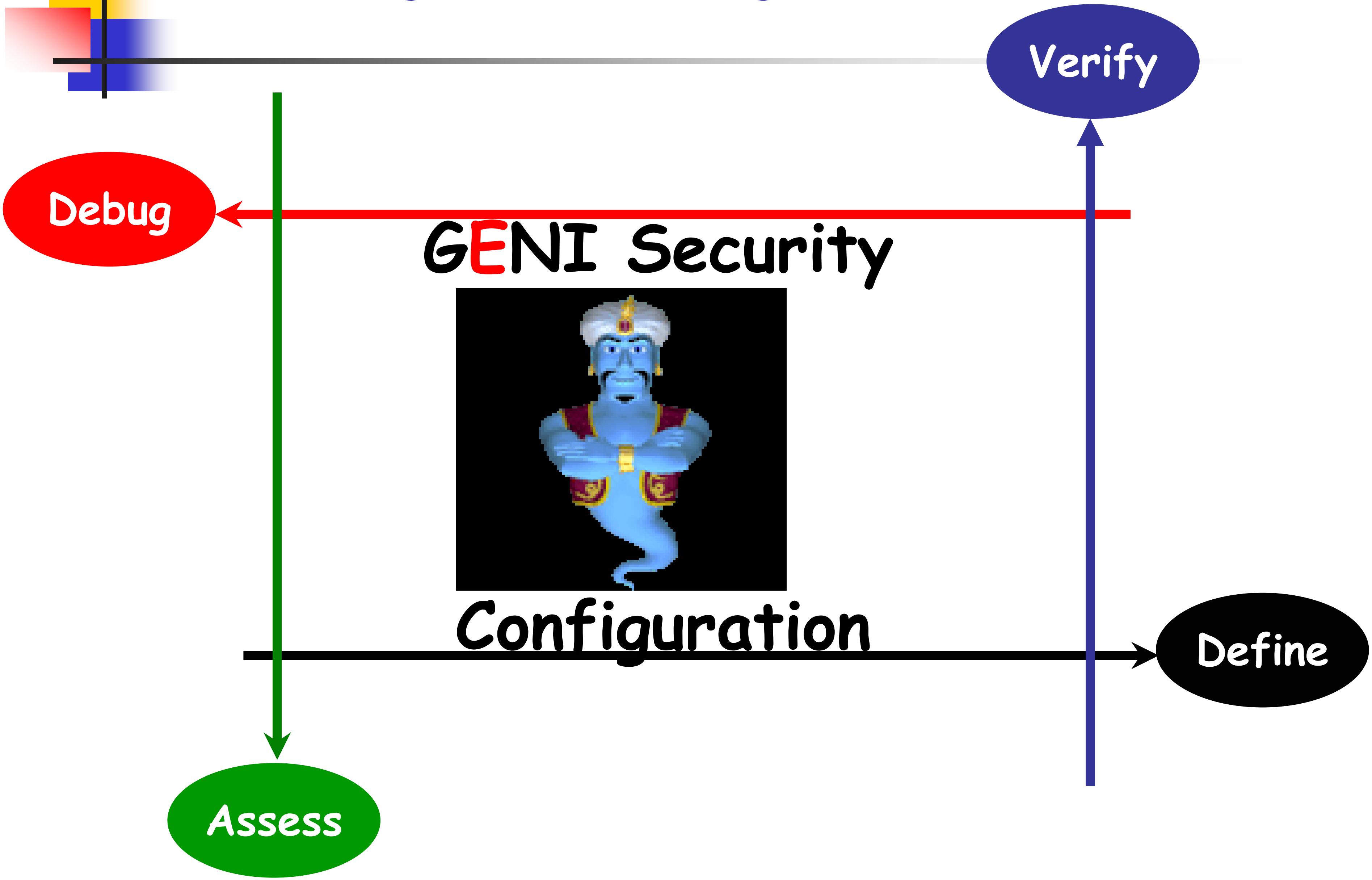
GENI Challenges

- Distributed resources
- Distributed control
- Dynamic policy coordination, interaction/federation, adaptation
- But still the goal is to keep it programmable, usable, assurable, and consistent → complex configuration

- How to provide end-to-end security configuration assurability/provability?
- How to make security systems configuration usable: high-level, distribution transparency?
- How to measure and assess configuration in term of risk, privacy, flexibility and cost?

© Ehab Al-Shaer

Putting GENI Configuration in a Box



Idea#1: ConfigChecker & ConfigLego– Automated Security Configuration Verification

- **Goals**
  - Global end-to-end unified verification across heterogeneous devices: unifying the representation of the security configurations of all network devices.
  - Integrating network and host security configuration checking: having a single model that can analyze both network and application level devices and services is the main focus.
  - Abstraction and Composablility
  - Scalability (10,000 of nodes)
- **Approaches**
  - Bottom-up
  - Modeling configuration semantic using Binary Decision Diagrams (BDD) gives canonical representation regardless of the syntax
  - **ConfigChecker**: models the network as a giant sate machines and used model checker and CTL to query and verify security configuration
    - Modeling packet transformations is an increasingly hard task.
    - Problems on a network-wide scale are impossible to detect manually, and automated tools focus on a single device or devices of a single type.
  - **ConfigLego**: allows for abstracting and composing portions of the network under-investigation

Modeling Access Control Policies

- **Single-trigger policy** is an access policy where only one action is triggered for a given packet. **C_i is the** 1st **match leads to action *a***

$$P_a = \bigvee_{i \in index(a)} (\neg C_1 \wedge \neg C_2 \ldots \neg C_{i-1} \wedge C_i)$$

$$P_a = \bigvee_{i \in index(a)} \bigwedge_{j=1}^{i-1} \neg C_j \wedge C_i$$

- **Multiple-trigger policy** is an access policy where multiple different actions may be triggered for the same packet. **C_i is** any **match leads to action *a***

$$P_a = \bigvee_{i \in index(a)} C_i$$

where $index(a) = \{i \mid R_i = C_i \rightsquigarrow a\}$

Intra-Policy Conflicts Formalization : Crypto-access List

- Policy expression S_a represents a policy that incorporates rule R_i, and S'_a is the policy with R_i excluded. R_i may be involved in the following conflicts:

  - **Shadowing:**

    $[(S'_{a_i} \Leftrightarrow S_{a_i}) = true]$ and $[(C_i \Rightarrow S'_{a_i}) = false]$

  - **Redundancy:**

    $[(S'_{a_i} \Leftrightarrow S_{a_i}) = true]$ and $[(C_i \Rightarrow S'_{a_i}) \neq false]$

  - **Exception:**

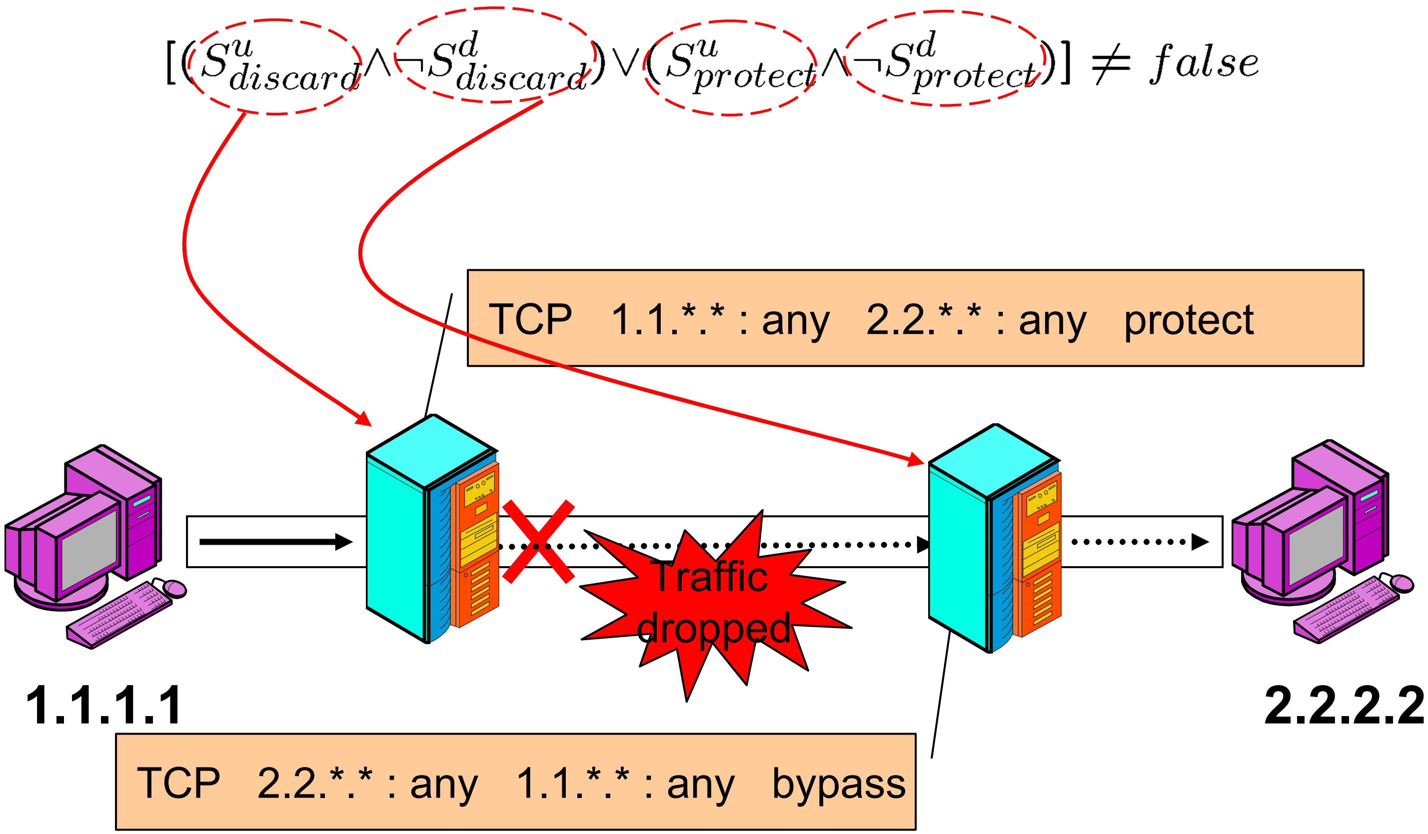
    $[(S'_{a_i} \Leftrightarrow S_{a_i}) \neq true]$ and $[(C_i \Rightarrow S'_{a_i}) = false]$

  - **Correlation:**

    $[(S'_{a_i} \Leftrightarrow S_{a_i}) \neq true]$ and $[(C_i \Rightarrow S'_{a_i}) \neq false]$

IPSec Inter-Policy Conflicts Formalization: Crypto-access Lists

- Shadowing: **upstream policy blocks traffic**



Diagnosing Unreachablility Problems between Routers and Firewalls

- **Flow-level Analysis:** Is flow C_k forwarded by routers in *L* (each of routing tables BDD T^i_j for router *i* and port j) but Blocked due to conflict between *Routing* and *FW Filtering*:

$$[(C_k \Rightarrow \bigwedge_{(i,j)\in L} T^i_j) \wedge (C_k \Rightarrow \neg S^n_A)] \neq false$$

  - This shows that a traffic C_j is forwarded by the routing policy, T'_j, from node *i* to *n* but yet blocked by the filtering policy, $S^n_{discard}$, of the destination domain.
- **Path-level Analysis:** Discovering Any Unreachability Conflicts between *Routing* and *Filtering*:

$$\phi_k \leftarrow [SAT(\bigwedge_{(i,j)\in path(x)} T^i_j \wedge \neg S^n_A \wedge \neg(\bigwedge_{i=1,k-1} \phi_i))] \neq false$$

  - For phi=1, n misconfiguration examples, and phi(0) = ture
- **Network or Federated-level Analysis:** Spurious conflict between downstream *d* and upstream *u* ISP domains:

$$[(S^u_{bypass} \wedge \neg S^d_{bypass}) \vee (S^u_{limit} \wedge S^d_{discard})] \neq false$$

  - Notice that $S_{discard}$, S_{bypass} and S_{limit} are filtering policies representations related to the filtering actions as described in [ICNP05, CommMag06].

ConfigChecker Queries (Model Checker approach)

- **Q1: Reachablility Soundness:**
  - From any source node *ip1* if there is a next-hop to destination *ip2*, then there must be a way that eventually leads to *ip2* from *ip1*.

$$Q = (loc(ip1) \wedge EX(dest = ip2)) \rightarrow loc(ip1) \wedge EF(dest(ip2) \Leftrightarrow loc(ip1))$$

- **Q2: Discovering Broken End-to-end IPSec Tunnel:**
  - Given a specific flow, will it stay in a tunnel until the final destination? (assuming the IPSec gateways are a hop away from the source and destination)

$$Q = (src = a1 \wedge dest = a2 \wedge loc(a1) \wedge IPSec(encT)) \rightarrow \mathbf{AU}((IPSec(encT) \vee loc \rightarrow \mathcal{G}), loc(a2))$$

- **Q3: What nodes have access to the plain-text packet:**
  - Given a specific flow, which nodes will eventually have access to the packet without encryption?

$$Q = AF\_(flow(ip1, ip2) \wedge loc(ip1)) \wedge \neg IPSec(encrypt)$$

ConfigChecker Queries

- **Q4: Back-door access after route changes:**
  - What is difference in the new configuration as compared with the ordinary original one. Is there any backdoor?

$$C_{org} \triangleq [\neg multiroute \wedge src = a1 \wedge dest = a2 \wedge loc(a1) \rightarrow AF(loc(a2) \wedge src = a1 \wedge dest = a2)]$$

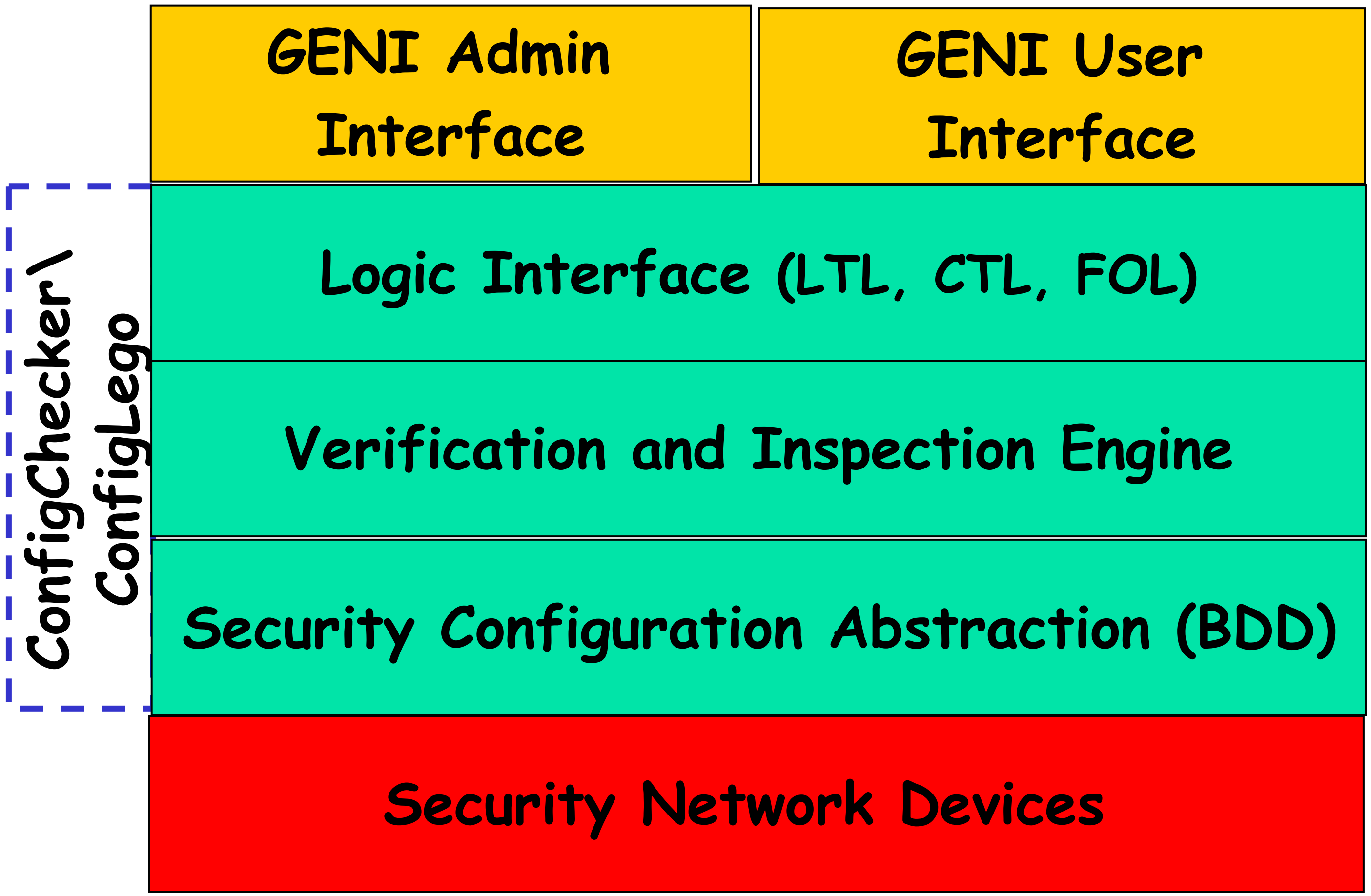
$$C_{new} \triangleq [multiroute \wedge src = a1 \wedge dest = a2 \wedge loc(a1) \rightarrow AF(loc(a2) \wedge src = a1 \wedge dest = a2)]$$

$$Backdoors: \neg C_{org} \wedge C_{new}$$

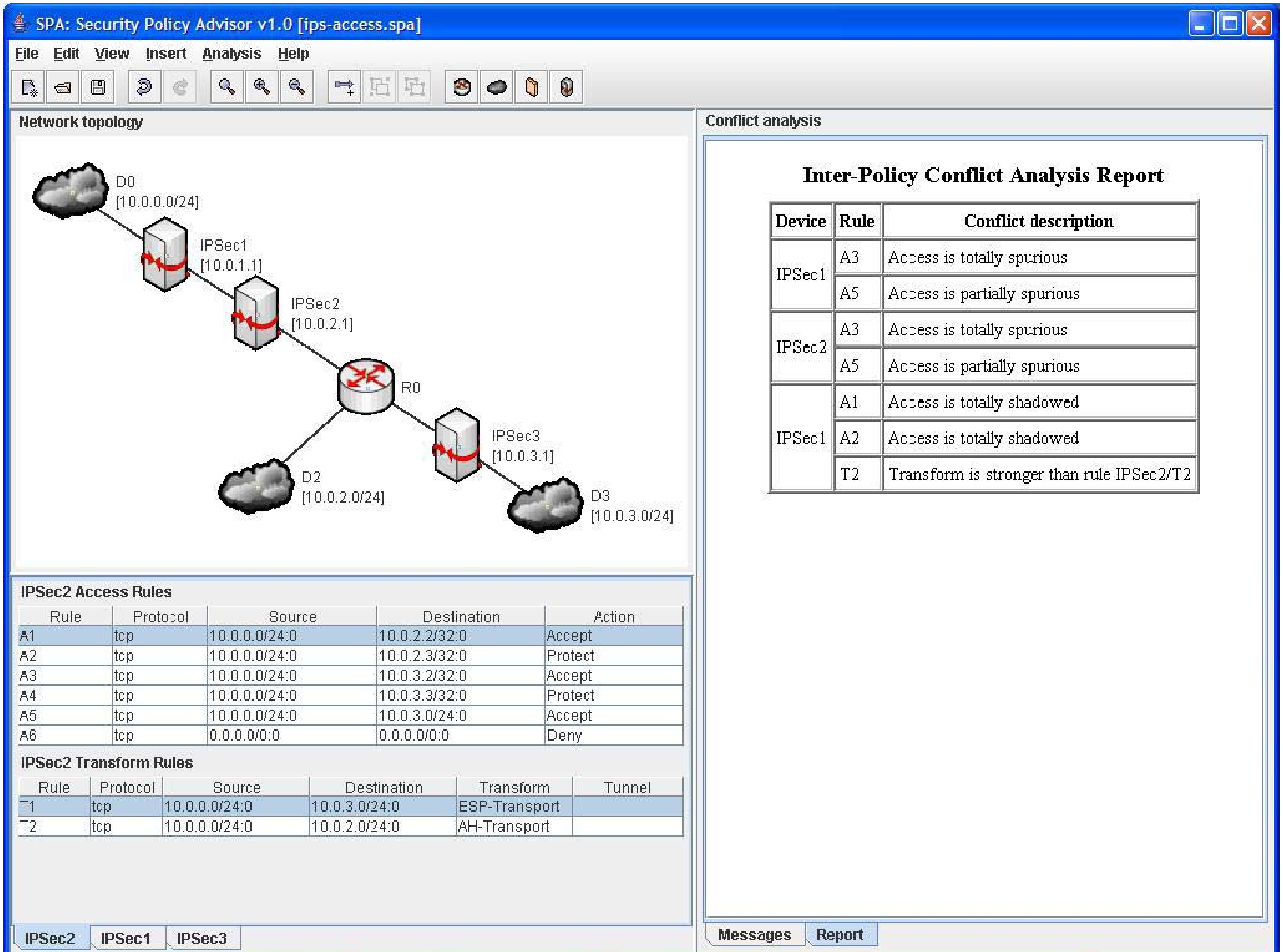
$$Broken\ flows: \neg C_{new} \wedge C_{org}$$

More information on ConfigChecker: www.arc.depaul.edu

Idea#1: GENI ConfigChecker / ConfigLego



Policy Advisor Tool for Distributed Policy (Firewall & IPSec) Management

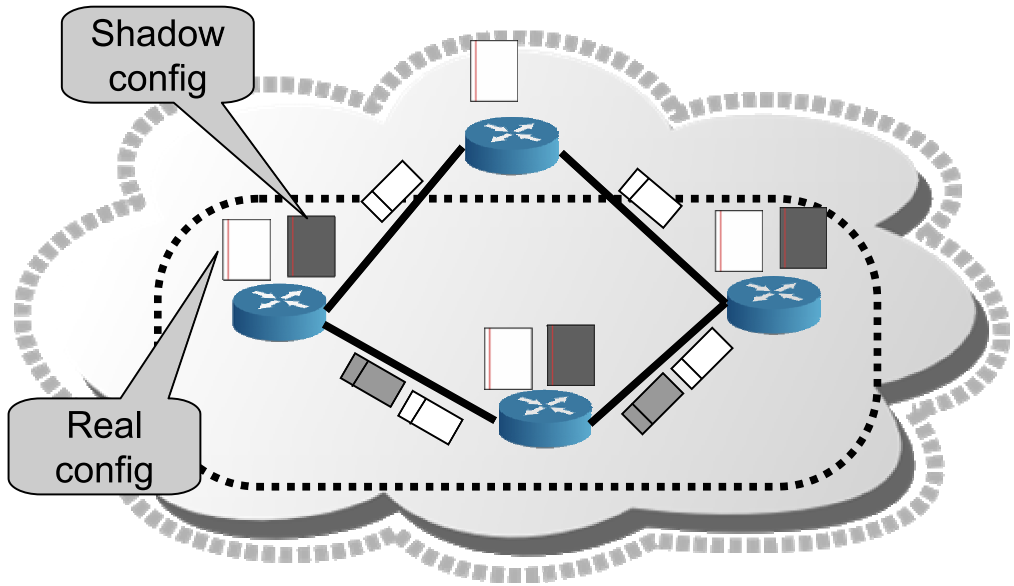


Intra-Policy Advisor Tool is used by the following 43 companies and institutions as of November, 2006

- Lisle Technology Partners, USA;
- Phontech, Norway;
- Naval Surface Warfare Center, Panama City, USA;
- Cisco Systems, USA;
- AT&T, USA;
- Gateshead Council, UK;
- ISRC, Queensland University of Technology, Australia;
- Imperial College and UCL, London, UK;
- Danet Group, Germany;
- TNT Express Worldwide, UK Ltd, United Kingdom;
- Checkpoint, USA;
- FireWall-1, The Netherlands;
- UFRGS, Brasil;
- DataConsult, Lebanon;
- Rosebank Consulting, GB;
- Columbia University, USA;
- Mayer Consulting, USA;
- Panduit Corp, USA;
- UPMC Paris 5 University, France;
- Royal institute of Science, Sweden;
- GE, US;
- Aligo, USA.
- Others not listed

Idea#2: Shadow Configurations for On-line Configuration Debugging

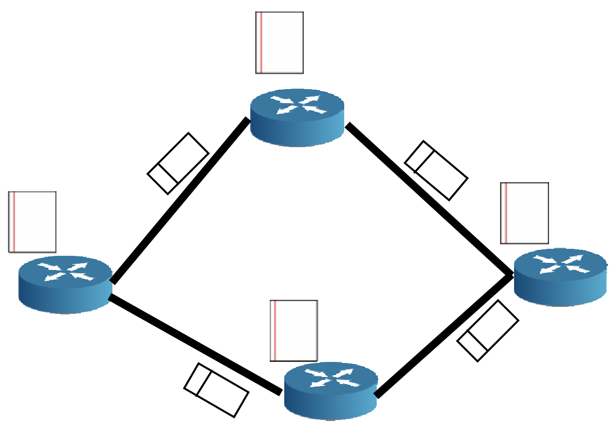
- Use Deployed Network
- Allow an additional shadow configuration on each router
  - Routing, ACLs, interface addresses, etc.
- Scalable and realistic (no modeling)
- Two key capabilities
  - Pre-deployment testing/debugging
  - Does not affect real traffic



Scenario: Config Changes

- **Scenario: Change configuration parameters**
  - Address performance/security issues
  - Deploy new services (e.g., filters, IDS probes and QoS)
- **Operation**
  1) Copy real traffic to shadow plan
  2) Change shadow and test
  3) Store and aggregate traces
  4) Debug, compare and isolate
  5) Commit real and shadow

- **Prototype for Routing only (with Richard Wang, Yale) – see SIGCOMM 2008**



Summary & Future Work

- **GENI success will be greatly dependant on assurability and usability of security configuration: define, verify, evaluate/ metrics and optimize**
- **Other Issues**
  - How integrate application level and network level access control
  - How to build API and high-level user interfaces to help using the underlying configuration engnes
  - Measuring security
  - Top-down approach: Balancing security, usability, privacy and cost

Thank You!!