Creating and Maintaining Firewall Policies

An Approach Based on Argumentation and its Empirical Evaluation

Nirav Ajmeri
Advisor: Munindar P. Singh

Department of Computer Science
North Carolina State University

December 2, 2014

Outline

Introduction

ArgPol
- Approach
- Argumentation Roadmaps
- Realizing Requirements through ArgPol

Empirical Evaluation
- Design
- Results

Conclusion

Defining and Maintaining Policies

- Security policies involve
  - Complex interdependencies
  - Conflicting user requirements
- Anomalies
- Maintenance is a challenge
- Proposal: Apply argumentation to
  - Capture design rationale
  - Reason about policies

Firewalls

Example of firewall policy

#	Action	Protocol	Source	Port
1	Allow	*	*	20
2	Allow	*	*	80
3	Block	*	locA.example.com	20
4	Allow	*	*	21
5	Block	*	*	53
6	Allow	TCP	locB.example.com	23
7	Block	*	*.example.com	*
8	Allow	UDP	locB.example.com	5027
9	Allow	UDP	*	*
10	Block	*	*	6889
11	Allow	*	example.net	53
12	Block	*	*	*

Firewalls

Conflicts and redundancies

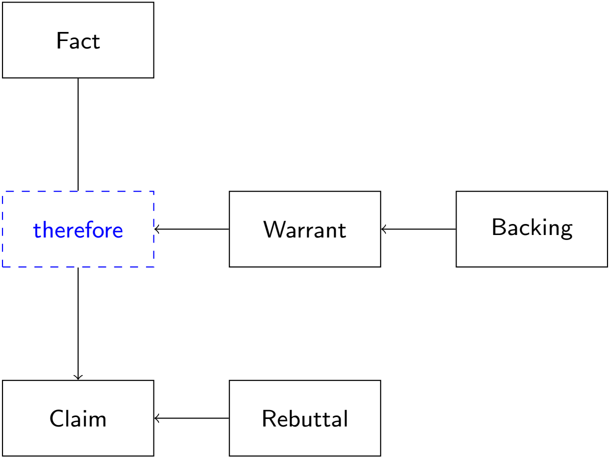
#	Action	Protocol	Source	Port
1	Allow	*	*	20
2	Allow	*	*	80
3	Block	*	locA.example.com	20
4	Allow	*	*	21
5	Block	*	*	53
6	Allow	TCP	locB.example.com	23
7	Block	*	*.example.com	*
8	Allow	UDP	locB.example.com	5027
9	Allow	UDP	*	*
10	Block	*	*	6889
11	Allow	*	example.net	53
12	Block	*	*	*

Argumentation

- Argument is constructed from a set of statements
  - Consists three parts
    - Conclusion (or, equivalently, the claim)
    - Set of premises (or, equivalently, the support)
    - Inference from the premises to the conclusion
  - Supported or attacked by other arguments
  - Traditionally represented as a pair $\langle \text{premise}, \text{conclusion} \rangle$
- Argumentation involves constructing chain of arguments
  - Conclusion of one inference is a premise of the next one
- No well established measure of completeness

Argument

Based on Stephen Toulmin



Argumentation Schemes

- Patterns for constructing arguments
- Decision-maker follows an argumentation scheme to iteratively collect evidence and infer the veracity of the conclusion
- Provide a set of *critical questions* for evaluating if the argument holds
  - Choosing right critical question is nontrivial
- Represented as $\langle \text{premise}, \text{conclusion}, \text{questions} \rangle$

Outline

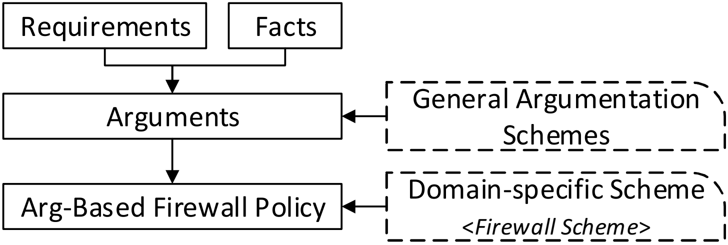
Introduction

ArgPol
- Approach
- Argumentation Roadmaps
- Realizing Requirements through ArgPol

Empirical Evaluation
- Design
- Results

Conclusion

The ArgPol Approach



- Using general argumentation schemes, create arguments from requirements and facts
- Synthesize requirement-level arguments using the firewall argumentation scheme
- Incorporate evidence
- Argumentation roadmap for completeness

Requirements for the Policy

Example requirements

#	Description
R	Example Inc. requires file transfer
F	FTP and SFTP enable file transfer
R	Example Inc.'s public website is to be made available via port 80
R	Prevent known attacks
F	There is an attack history from locA.example.com through port 20
R	Example Inc. requires telnet access
F	Telnet requires access to port 23
R	Existing application requires access to UDP port 5027
R	Example Inc. wants to prevent access to torrent
F	Torrent uses port 6889
R	Enable DNS for example.net
F	DNS requires access to port 53

Practical Reasoning Scheme (by Walton)

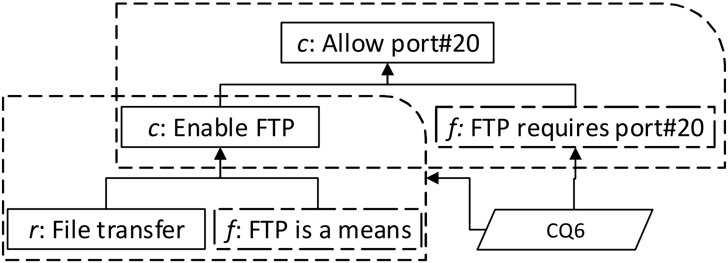
- Premise (Major) h_1: R is a requirement.
- Premise (Minor) h_2: Action A is a means to realize R.
- Conclusion c: A should be carried out.

Critical questions

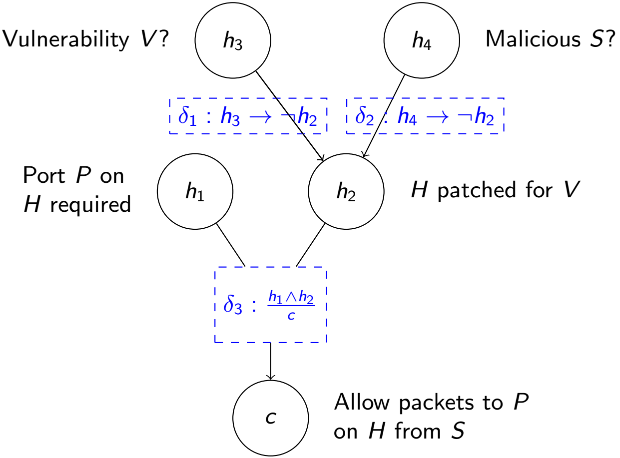
- CQ1. What other requirements that might conflict with R should be considered?
- CQ2. Are there alternative means available for carrying out R?
- CQ3. Among bringing about A and these alternative actions, which is the most efficient?
- CQ4. Is it practically possible to bring about A?
- CQ5. What consequences of bringing about A should also be taken into account?
- CQ6. Are other actions, in addition to A, required to bring about R?

Argument for the FTP Requirement

Using practical reasoning scheme by Walton



Firewall Argumentation Scheme



Firewall Argumentation Scheme

Critical Questions

CQ1. Is there any requirement associated with port P?

CQ2. Are there any known vulnerabilities V?

CQ3. Is there any evidence that host H is updated to handle known security vulnerability V?

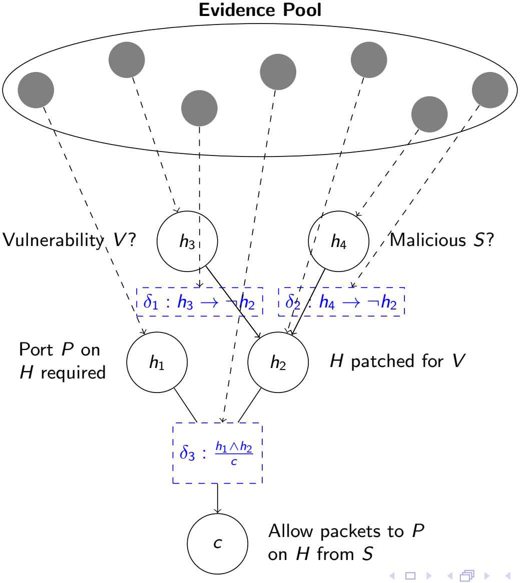
CQ4. Is there any evidence that source S is malicious?

Evidence-Based Argumentation

Traditional argumentation disregards degree of belief

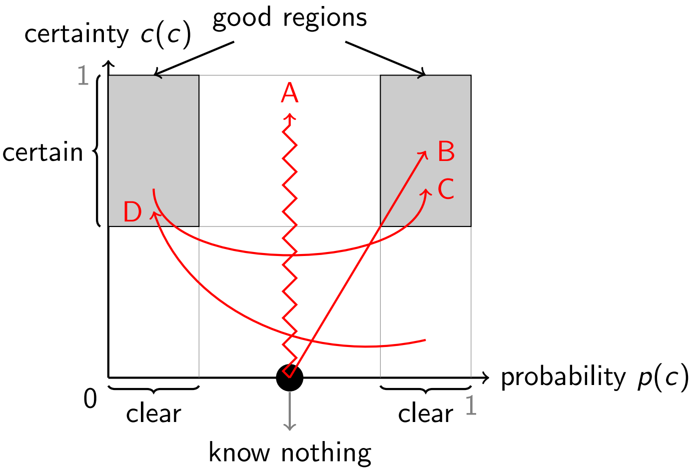
- Premises and inference rules share evidence
- A belief can be expressed as a triple $\langle \text{belief}, \text{disbelief}, \text{uncertainty} \rangle$
  - Computed from evidence

Incorporating Evidence in Firewall Argumentation Scheme



Argumentation Roadmaps

Probability-certainty paths



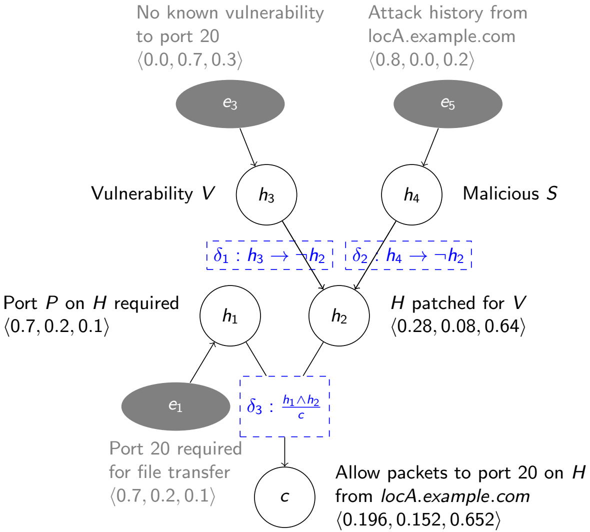
Realizing Requirements through ArgPol

Pieces of evidence corresponding to the file transfer requirement

Evidence	Premise	Belief Measure	Description	Related Argument
e_1	h_1	$\langle 0.7, 0.2, 0.1 \rangle$	Port 20 is required for file transfer via FTP	$A1$
e_2	h_1	$\langle 0.8, 0.1, 0.1 \rangle$	Port 21 is required for secured file transfer via SFTP	$A2$
e_3	h_3	$\langle 0.0, 0.7, 0.3 \rangle$	No known vulnerability to port 20	$A1$
e_4	h_3	$\langle 0.0, 0.7, 0.3 \rangle$	No known vulnerability to port 21	$A2$
e_5	h_4	$\langle 0.8, 0.0, 0.2 \rangle$	Attack history from locA.example.com through port 20	$A1$
e_6	h_4	$\langle 0.0, 0.9, 0.1 \rangle$	No attack history on port 21	$A2$
e_7	δ_1	$\langle 0.8, 0.0, 0.2 \rangle$	Decision-maker's experience in the rule	$A1$, $A2$
e_8	δ_2	$\langle 0.8, 0.0, 0.2 \rangle$	Decision-maker's experience in the rule	$A1$, $A2$

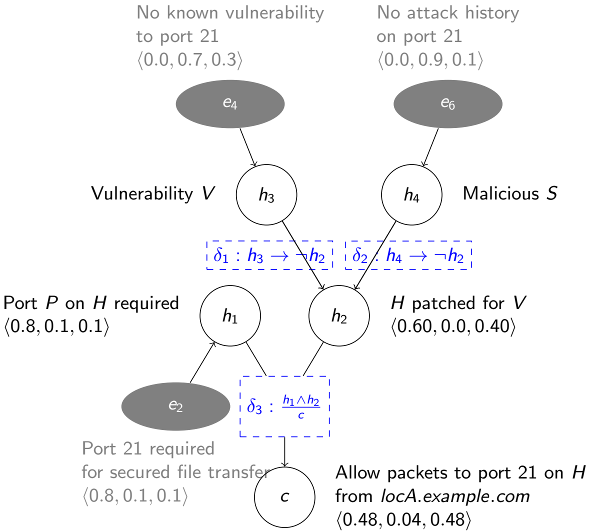
Realizing Requirements through ArgPol

Argument A1: Allow packets to port 20 from locA.example.com

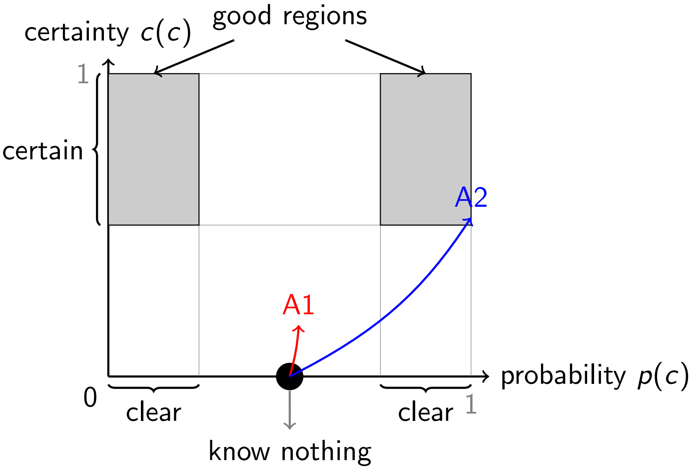


Realizing Requirements through ArgPol

Argument A2: Allow packets to port 21 from locA.example.com



Probability-certainty paths for argument $A1$ and $A2$



Outline

Introduction

ArgPol
- Approach
- Argumentation Roadmaps
- Realizing Requirements through ArgPol

Empirical Evaluation
- Design
- Results

Conclusion

Design

Participants

- 24 computer science (21 graduate, and three undergraduate) students
  - More than three years of programming and software development experience
  - Familiarity with conceptual modeling and network security
    - 19 with industry or academic experience with network security
    - 16 with industry or academic experience with conceptual modeling
- Split in two balanced groups A and B

Design

Study mechanics and deliverables

- One factor design with two alternatives
- Three phased human subject study
  - Phase 1: Learn one of the two approaches, and design solution for an academic scenario
  - Phase 2: Design solution for an industry scenario
  - Phase 3: Make changes to an existing solution
- Deliverables
  - Group A: Define firewall packet filtering rules based on the requirements
  - Group B: Create arguments using schemes and critical questions

Response Variables

Learnability. Time in minutes to learn and design the solution. Lower is better.

Maintainability. Time in minutes to make changes to the solution. Lower is better.

Completeness. Ratio of the number of requirements satisfied to the total number of requirements in the design and the maintenance phases. Higher is better.

Correctness. Ratio of the number of requirements satisfied to the number of requirements attempted. Higher is better.

Quality. Product of *completeness* and *correctness*. Higher is better.

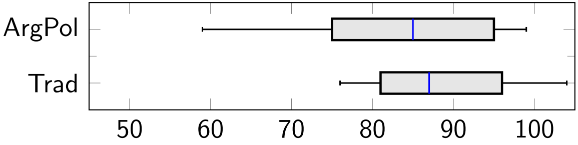
Difficulty to learn. Difficulty rating by participant on scale of 1–5 interpreted as very easy, easy, neutral, difficult, and very difficult. Lower is better.

Difficulty to apply. Difficulty rating by participant on scale of 1–5 interpreted as very easy, easy, neutral, difficult, and very difficult. Lower is better.

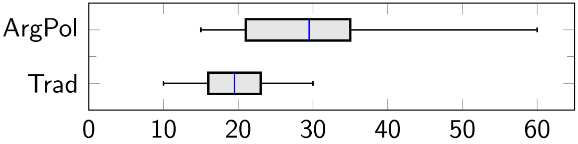
Effort. Product of time in minutes to design the solution, and *difficulty to apply*. Lower is better.

Effort÷Completeness. Product of the number requirements satisfied and *effort*, divided by *completeness*. Lower is better.

Learnability and Maintainability



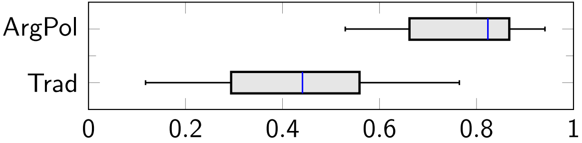
Time to learn, $\mu_{ArgPol} < \mu_{Trad}, p = 0.45$



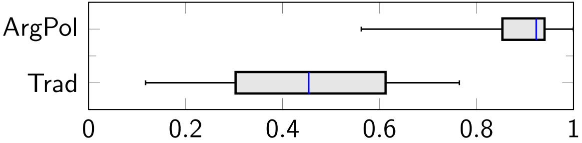
Time to maintain, $\mu_{Trad} < \mu_{ArgPol}, p = 0.03$

Completeness and Correctness

Measured for Phases 2 and 3

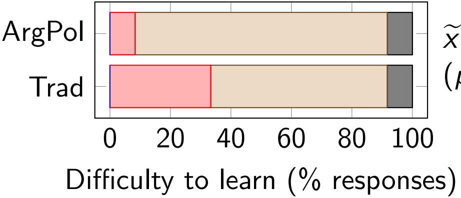


Completeness, $\mu_{ArgPol} > \mu_{Trad}, p < 0.01$

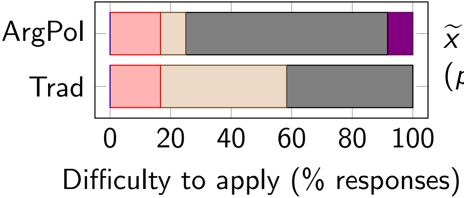


Correctness, $\mu_{ArgPol} > \mu_{Trad}, p < 0.01$

Subjective Difficulty to Learn and Apply



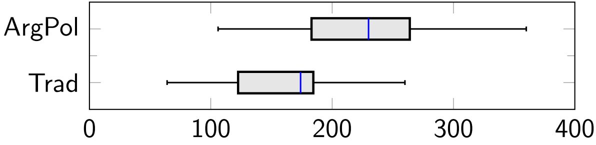
$\widetilde{x}_{Trad} < \widetilde{x}_{ArgPol}$
$(p = 0.2468)$



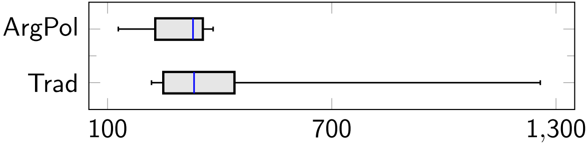
$\widetilde{x}_{Trad} < \widetilde{x}_{ArgPol}$
$(p = 0.1613)$

Effort and Effort÷Completeness

For Phases 2 and 3



Effort, $\mu_{ArgPol} > \mu_{Trad}$, $p = 0.02$



Effort÷Completeness, $\mu_{ArgPol} < \mu_{Trad}$, $p = 0.28$

Threats to Validity

- **Skill difference.** Two balanced groups of participants based on a pre-participation survey
  - Educational background
  - Prior experience with conceptual modeling
  - Prior experience with network security
  - Familiarity with firewall
- **Forgetting to report information.** Participant reported time and effort after each phase

Outline

Introduction

ArgPol
- Approach
- Argumentation Roadmaps
- Realizing Requirements through ArgPol

Empirical Evaluation
- Design
- Results

Conclusion

Conclusions and Future Work

- Support incorporating evidence into argumentation schemes
- Belief, disbelief, and uncertainty of evidence guides the argumentation process toward completeness
- Empirical evaluation indicates
  - ArgPol yields significantly better completeness, and correctness
  - No significant difference in learnability, maintainability and effort÷completeness
- In future, compare ArgPol with other argumentation-based approaches

Thank you

Formalization (1)

Definition 1 (Evidence)

Decision-maker has a finite set of evidence $E = \{e_1, \ldots, e_n\}$ such that

- $E = \{e_1, ..., e_n\}$ with $e_1, ..., e_n \in \mathcal{L}$ and $e_i \not\equiv e_j$ (for any $i \neq j$), and
- $m(e)$ is a probability *mass value* which satisfies a constraint:

$$m(e_1) + ... + m(e_n) = 1$$

Formalization (2)

Definition 2 (Belief Measure)

For a premise $h \in \mathcal{L}$ (or an inference rule δ), the belief measure: the belief $b(h)$, the disbelief $d(h)$, the uncertainty $u(h)$ of h:

$$b(h) = \Sigma_{\mathcal{I}(e_i) \subseteq \mathcal{I}(h)} m(e_i)$$
$$d(h) = \Sigma_{\mathcal{I}(e_i) \cap \mathcal{I}(h) = \emptyset} m(e_i)$$
$$u(h) = \Sigma_{\mathcal{I}(e_i) \cap \mathcal{I}(h) \neq \emptyset} m(e_i)$$

Equivalently, in the terms of logical entailment:

$$b(h) = \Sigma_{e_i \vdash h} m(e_i)$$
$$d(h) = \Sigma_{e_i \vdash \neg h} m(e_i)$$
$$u(h) = \Sigma_{e_i \nvdash h \text{ and } e_i \nvdash \neg h} m(e_i)$$

Formalization (3)

Definition 3 (Belief Measure Combination)

$$\text{The belief measure of } A \wedge B : \langle b_{A\wedge B}, d_{A\wedge B}, u_{A\wedge B}\rangle$$
$$= \langle b_A b_B, d_A d_B + d_A u_B + u_A d_B, 1 - b_{A\wedge B} - d_{A\wedge B}\rangle,$$

$$\text{The belief measure of } A \vee B : \langle b_{A\vee B}, d_{A\vee B}, u_{A\vee B}\rangle$$
$$= \langle (b_A + b_B)/2, (d_A + d_B)/2, (u_A + u_B)/2\rangle,$$

$$\text{The belief measure of } h_C : \langle b_C, d_C, u_C\rangle$$
$$= \langle b_\delta b_A, b_\delta d_A, 1 - b_C - d_C\rangle.$$

Formalization (4)

Definition 4 (Evidence Argument)

Evidence argument A consists of

- A set of premises $\Sigma_A \subseteq \Sigma$, each premise $h \in \Sigma_A$ has a belief measure $\langle b(h), d(h), u(h) \rangle$
- A set of inference rules $\Delta_A \subseteq \Delta$, each rule $\delta \in \Delta_A$ has a belief measure $\langle b(\delta), d(\delta), u(\delta) \rangle$
- A conclusion c with a belief measure $\langle b(c), d(c), u(c) \rangle$,

where c can be derived from Σ_A and Δ_A.

Formalization (5)

Definition 5 (Answer to a Critical Question)

Let $E_{b(h_i)} \subseteq E$, $E_{d(h_i)} \subseteq E$, and $E_{u(h_i)} \subseteq E$ be three disjoint sets of evidence such that,

$$b(h_i) = \sum_{e_i \in E_{b(h_i)}} m(e_i)$$

$$d(h_i) = \sum_{e_i \in E_{d(h_i)}} m(e_i)$$

$$u(h_i) = \sum_{e_i \in E_{u(h_i)}} m(e_i)$$

Formalization (6)

Definition 6 (Completeness of an Argument)

An argumentation is complete if and only if the certainty of the conclusion is higher than a threshold, and the probability is either higher than a high threshold or lower than a low threshold.

Results

	Trad	ArgPol	*p*
Learnability (in minutes) ∓	88.66	**84**	0.45
Maintainability (in minutes) ∓	**20.08**	29.5	**0.03**
Completeness (in %) ∓	35.76	**74.54**	$<$ **0.01**
Correctness (in %) ∓	39.99	**82.93**	$<$ **0.01**
Quality (in %) ∓	14.30	**61.82**	$<$ **0.01**
Learning difficulty (1–5) †	3	3	0.25
Applying difficulty (1–5) †	**3**	4	0.16
Effort ∓	**158.33**	225.42	**0.02**
Effort÷Completeness ratio ∓	466.09	**291.99**	0.28