The Mathematical Foundations for Mapping Policies to Network Devices

Dinesha Ranathunga*, Matthew Roughan*, Phil Kernick**, Nick Falkner*,

* University of Adelaide

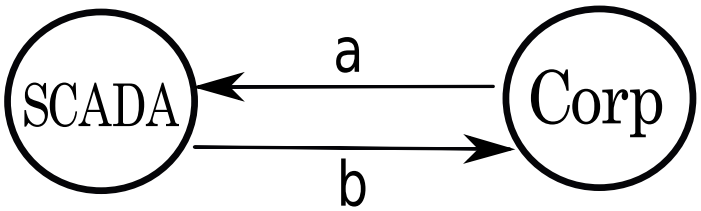
** CQR Consulting

Motivation

- Context is Policy Defined Networking (PDN)
- **Policy** and **Implementation** should be separate
- Then coupled back together (*i.e.*, policy mapped to devices)
- The coupling must be *formally* checkable

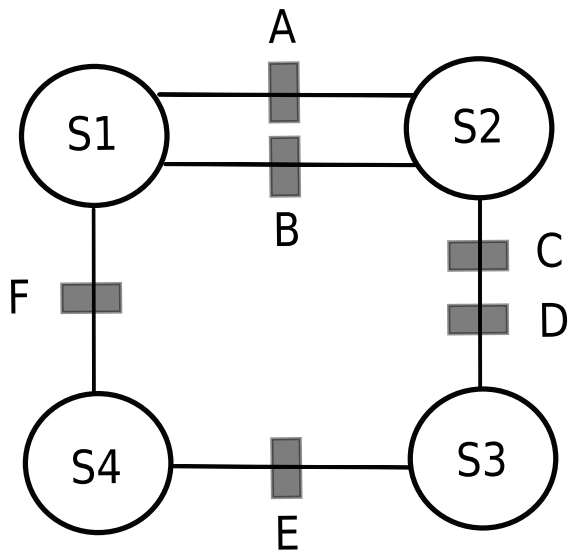
Example

Best-practice policy



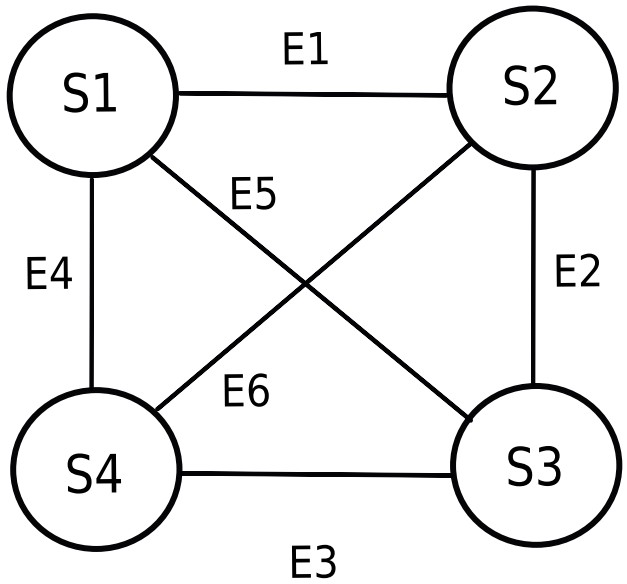
$\Rightarrow$

My network



Correct policy deployment is hard!

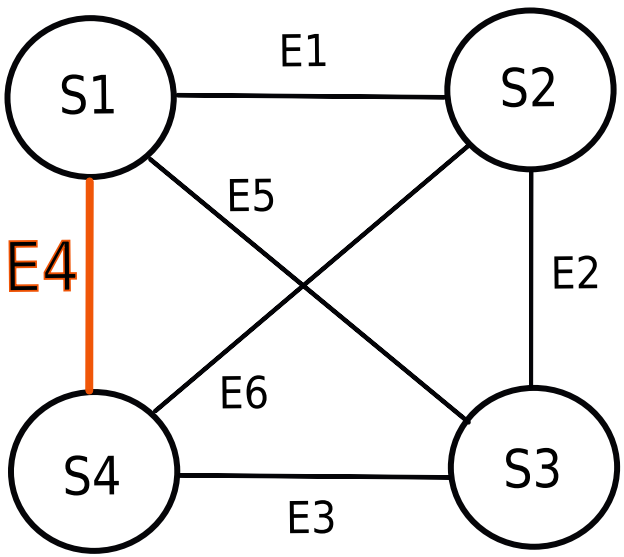
Policy graph



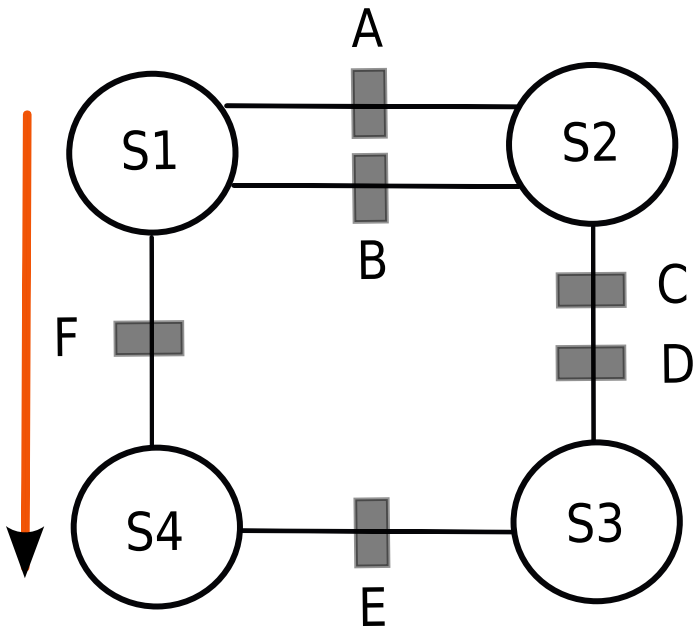
- *(endpoint-group, edge)* : commonly used to decouple policy from the network
  - *endpoint: e.g.,* a subnet, a user-group
  - *edge:* specifies relationship between endpoint-groups

Correct policy deployment is hard!

Policy graph



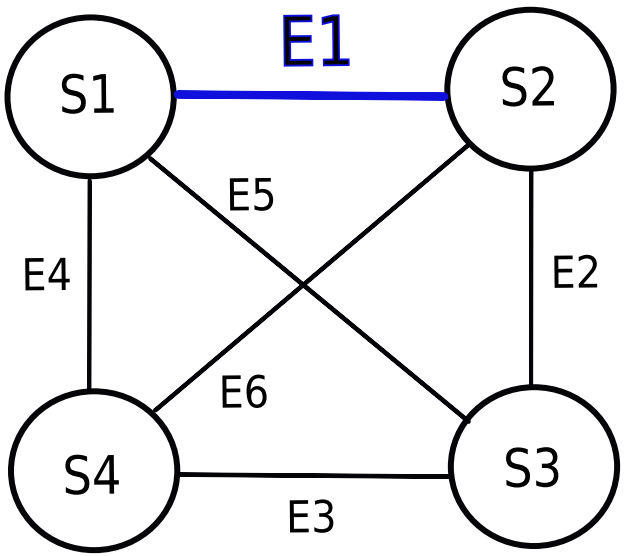
Network topology



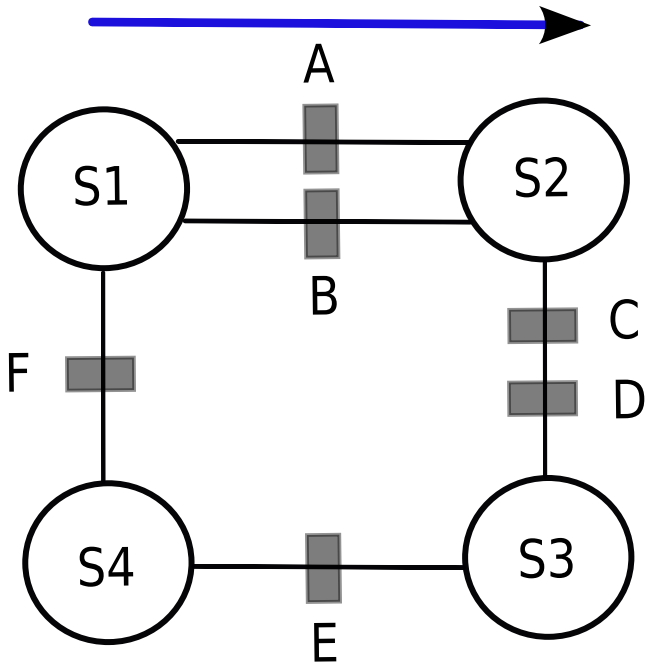
- *e.g.*, E4: $S1 \rightarrow S4$: *ssh allow*

Correct policy deployment is hard!

Policy graph



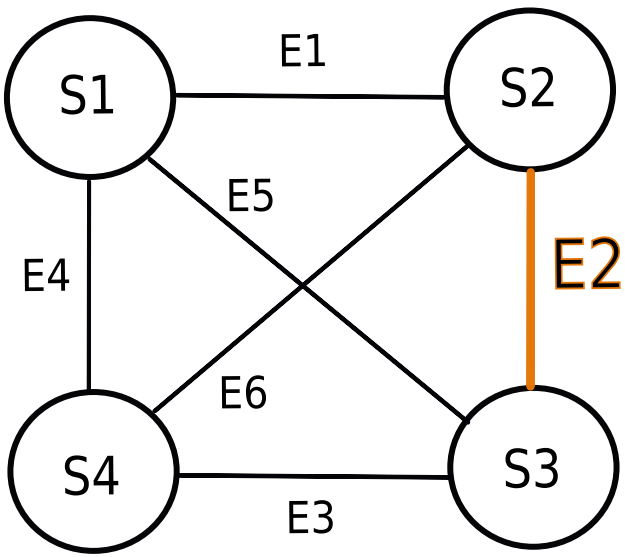
Network topology



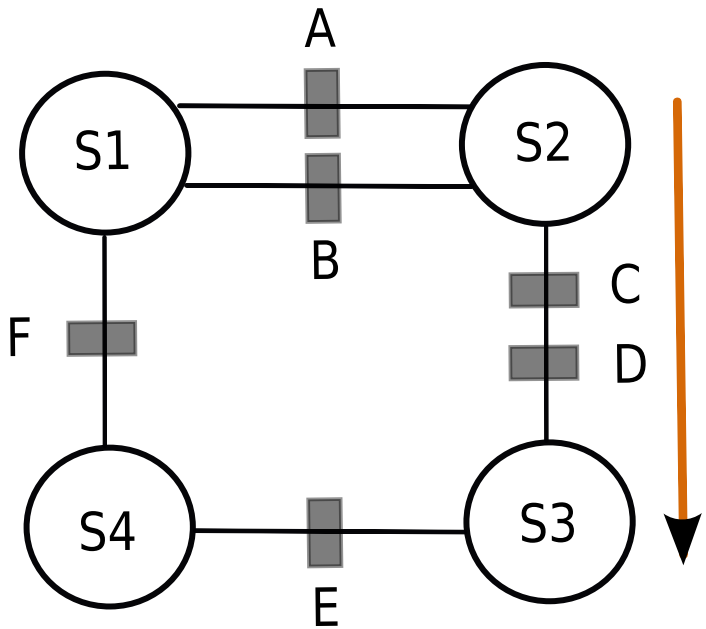
- *e.g.*, E1: $S1 \rightarrow S2$: *ssh allow*

Correct policy deployment is hard!

Policy graph



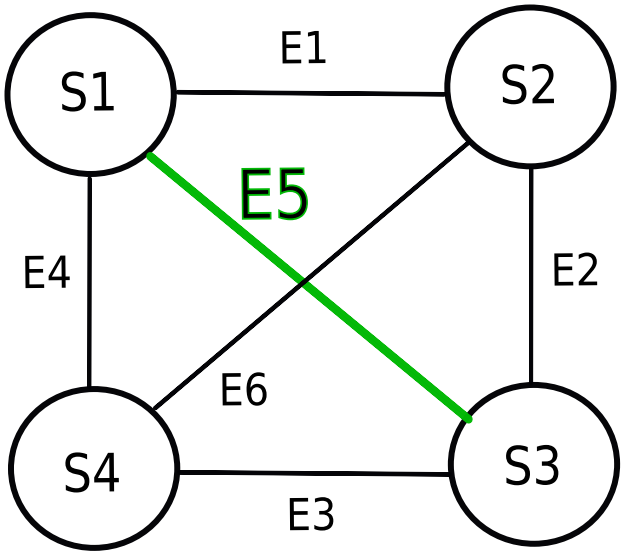
Network topology



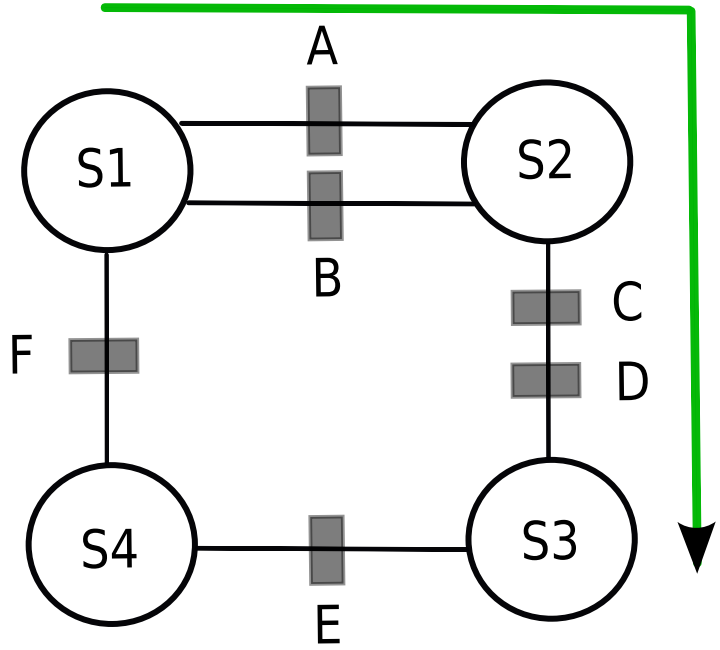
- *e.g.*, E2: $S2 \rightarrow S3$: *ssh allow*

Correct policy deployment is hard!

Policy graph



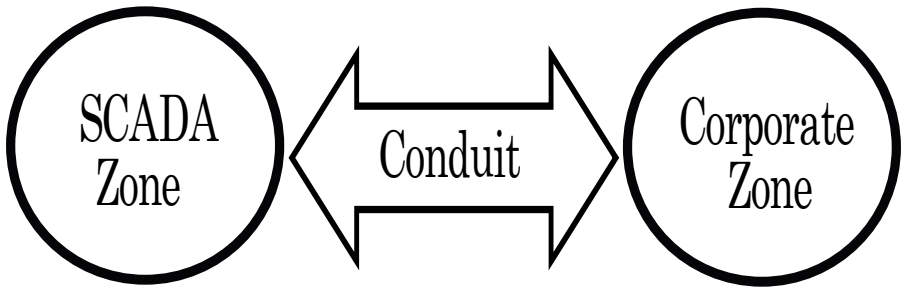
Network topology



- *e.g.*, E5: $S1 \rightarrow S3$: *ssh allow*

Existing standard for decoupling security policy from network

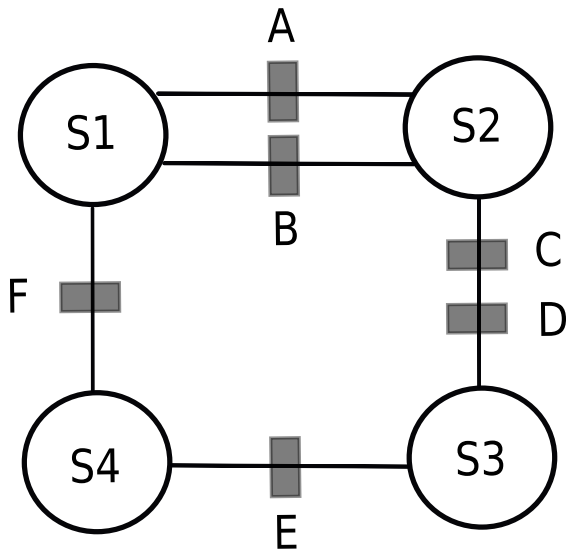
ANSI/ISA Zone-Conduit model [ANSI/ISA-62443-1-1]:



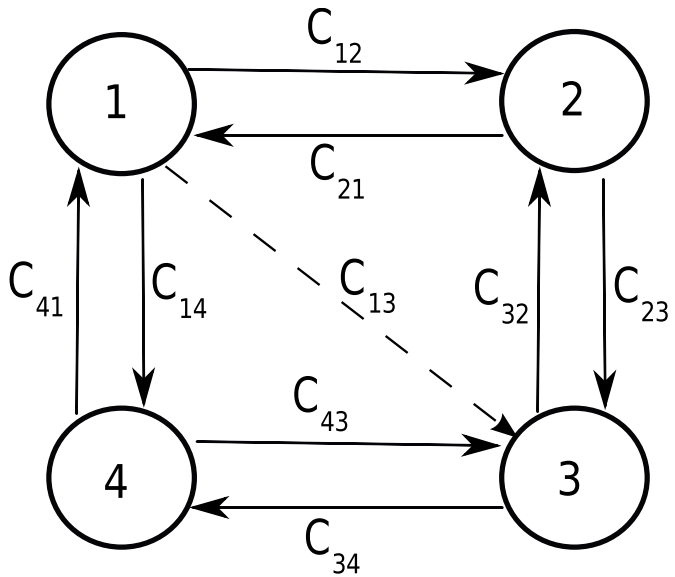
- Concrete instance of *(endpoint-pair, edge)* abstraction
- Allows to construct network-wide high-level security policy

Mapping security policy to firewalls: a simple example

(a) Network topology



(b) Zone-Conduit model of (a)



- *Primary* vs *Secondary* conduits
- How do we find all feasible primary- and secondary-conduits between zones?

Need a Mathematical approach

Why?

- Precision
- Unambiguity
- Verifiability
- Avoid redundant policy updates

What Maths in particular?

- Semiring algebra, why?
  - semiring properties allow lifting computations to a matrix and it converges
  - idea already used in meta-routing
- Consequences
  - policies need to adhere to semiring axioms
  - how policy should be described in a language

Implementation

- Computational limitation $O(n^4)$; n - number of zones
- n should be moderate
- We used it to map security control policies to real firewalls

Application to real SCADA case studies

SUC	Fire-walls	Zones	Max. hosts	ACLs	Average rules per ACL	Wrong firewall	Wrong interface	Wrong direction
1	3	7	67580	8	237	15	13	19
2	6	21	2794	12	16	3	2	5
3	4	10	886	8	6	2	1	4
4	3	9	2038	3	80	5	12	13
5	3	12	2664	12	677	15	8	26
6	3	13	3562	8	1034	21	15	19
7	6	15	3810	17	724	9	5	17

Conclusions

- Many obstacles to correct policy deployment in networks
- We address these challenges
  - network and vendor independent high-level policy semantics
  - generic algebraic framework to allocate policy to network devices
  - implementation that maps security policies to real firewalls

Bibliography

[1] C. J. Anderson et al. "NetKAT: Semantic foundations for networks". In: *ACM SIGPLAN Notices* 49.1 (2014), pp. 113–126.

[2] ANSI/ISA-62443-1-1. *Security for Industrial Automation and Control Systems Part 1-1: Terminology, Concepts, and Models*. 2007.

[3] Y. Bartal et al. "Firmato: A novel firewall management toolkit". In: *ACM TOCS* 22.4 (2004), pp. 381–420.

[4] E. Byres, J. Karsch, and J. Carter. "NISCC good practice guide on firewall deployment for SCADA and process control networks". In: *NISCC* (2005).

[5] J. D. Guttman and A. L. Herzog. "Rigorous automated network security management". In: *IJIS* 4.1-2 (2005), pp. 29–48.

[6] D. Ranathunga et al. "Identifying the Missing Aspects of the ANSI/ISA Best Practices for Security Policy". In: *1st ACM Workshop on Cyber-Physical System Security (CPSS)*. ACM. 2015, pp. 37–48.

[7] R. Soulé et al. "Merlin: A Language for Provisioning Network Resources". In: *ACM CoNEXT '14*. ACM. 2014, pp. 213–226.