Moving Target Defense for the Placement of Intrusion Detection Systems in the Cloud

Sailik Sengupta,
Subbarao Kambhampati

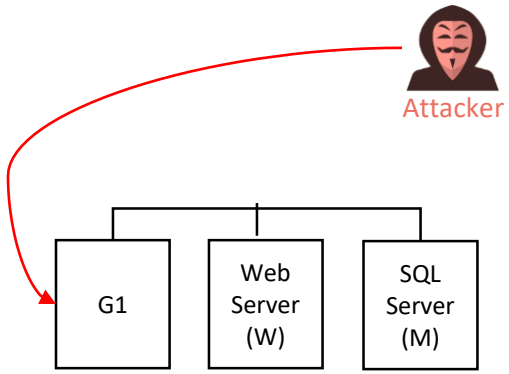
Yochan AI Lab

Ankur Chowdhary,
Dijiang Huang

SNAC

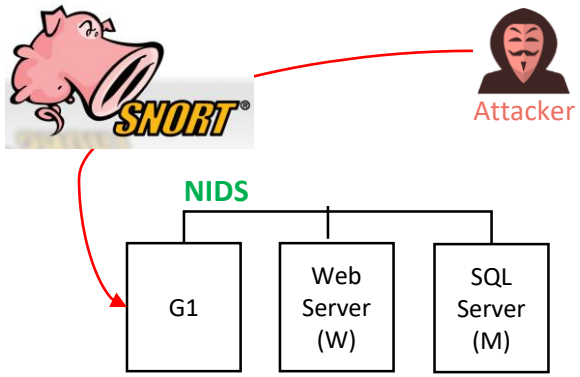
Secure Networking and Computing Lab

Intrusion Detection Systems?



Intrusion Detection Systems?

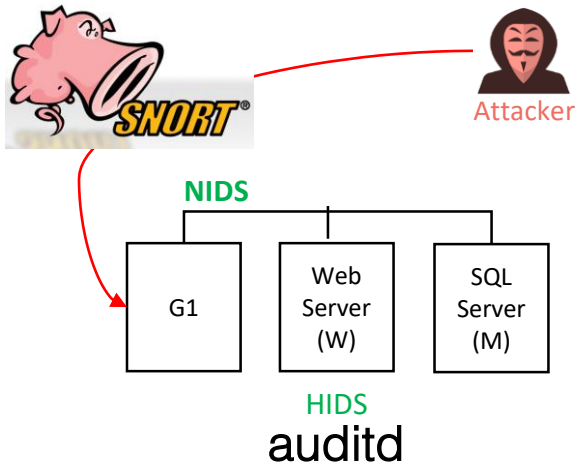
Network-Based Intrusion Detection Systems
- Checks payload on the network to infer if it is (going to be) malicious.



Intrusion Detection Systems?

Network-Based Intrusion Detection Systems
- Checks payload on the network to infer if it is (going to be) malicious.

Host-Based Intrusion Detection Systems
- Analyzes a computing system to detect anomalous behavior on it. Can monitor things from read/write access of files/folders to software calls that may record keystrokes etc.



Contents

- Motivation
- Problem Description
- Solution Methods
- Results
- Conclusions

Going All-out for Security in Large Cloud Networks

☹ NIDS increases
- Processing time of a packet
- Number of packets sent over the internal network

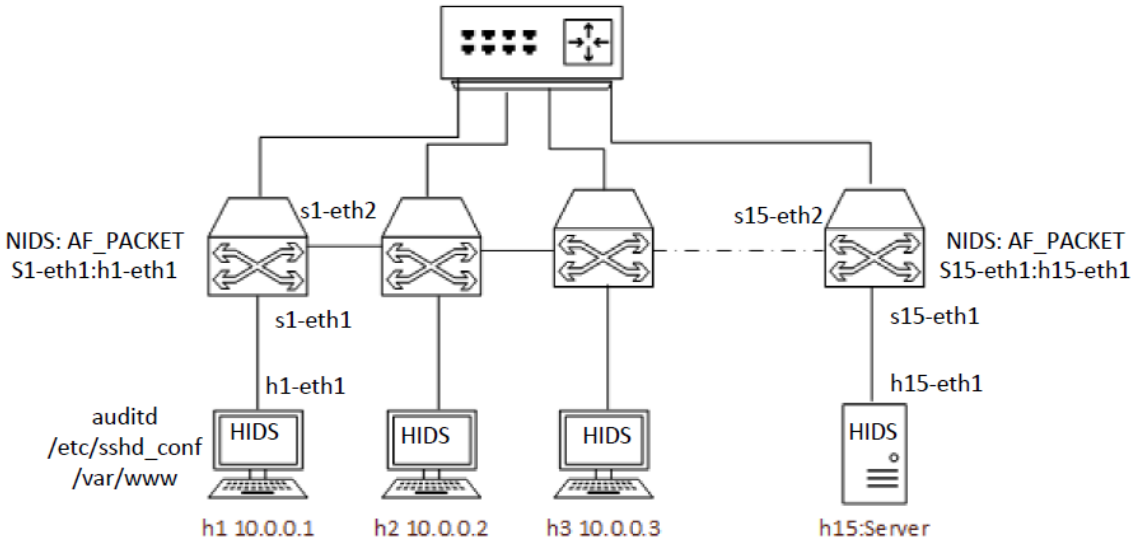
☹ HIDS increases
- Use of resources on a particular host etc.

Jha, S et al. 2002, Venkatesan, S et al. 2016

Going All-out for Security in Large Cloud Networks

☹ NIDS increases
- Processing time of a packet
- Number of packets sent over the internal network

☹ HIDS increases
- Use of resources on a particular host etc.

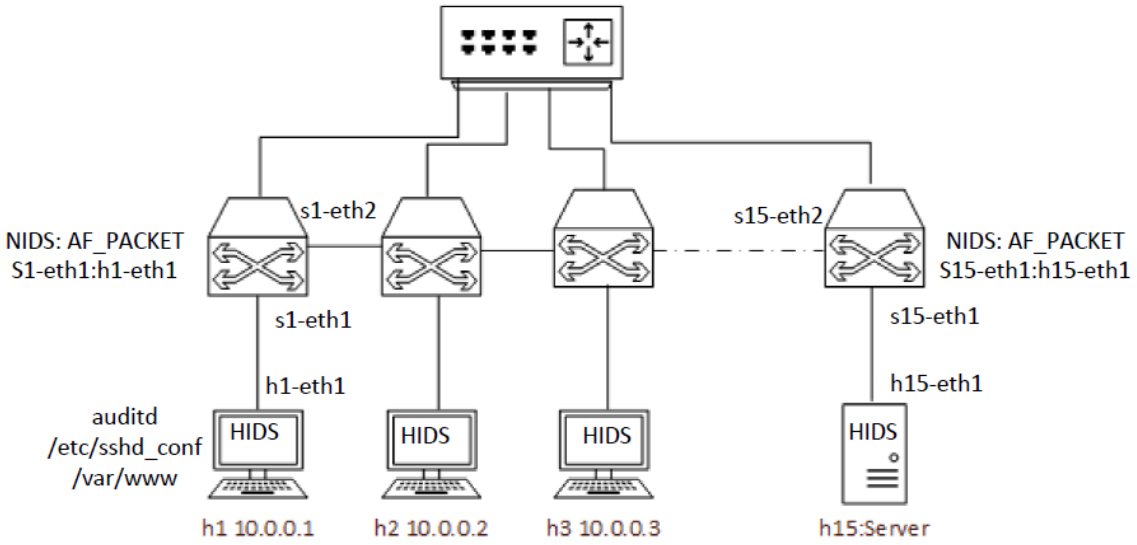


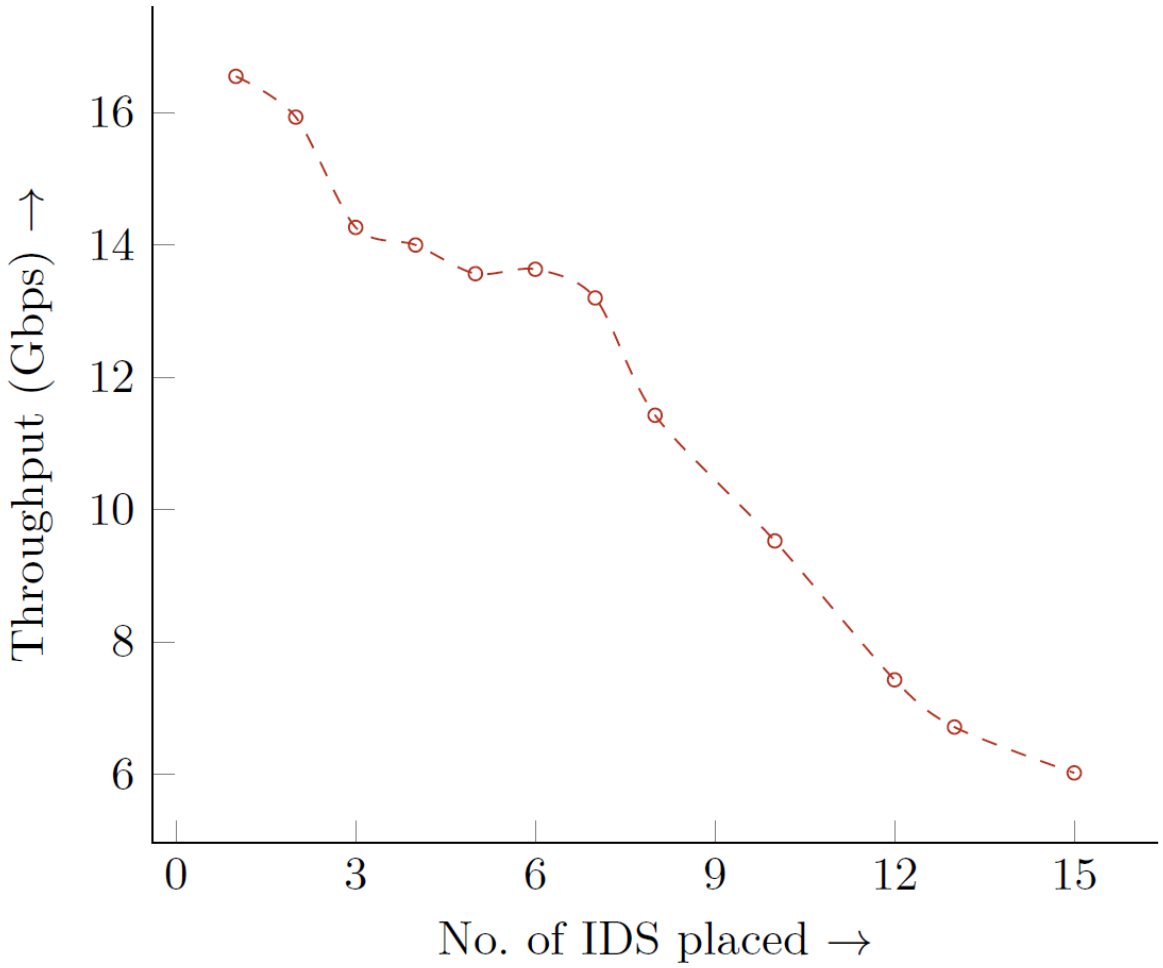
Jha, S et al. 2002, Venkatesan, S et al. 2016

Going All-out for Security in Large Cloud Networks

☹ NIDS increases
- Processing time of a packet
- Number of packets sent over the internal network

☹ HIDS increases
- Use of resources on a particular host etc.

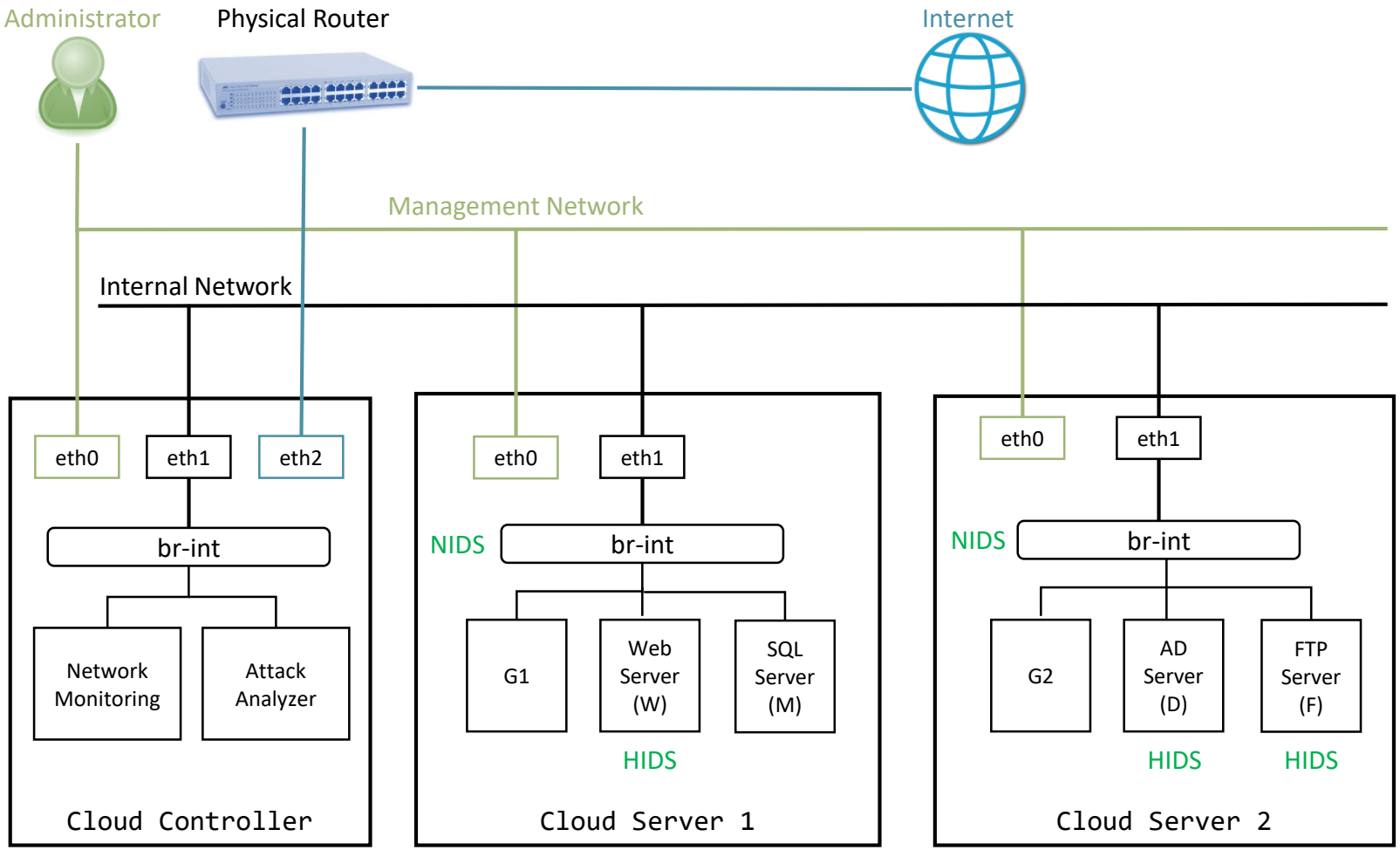




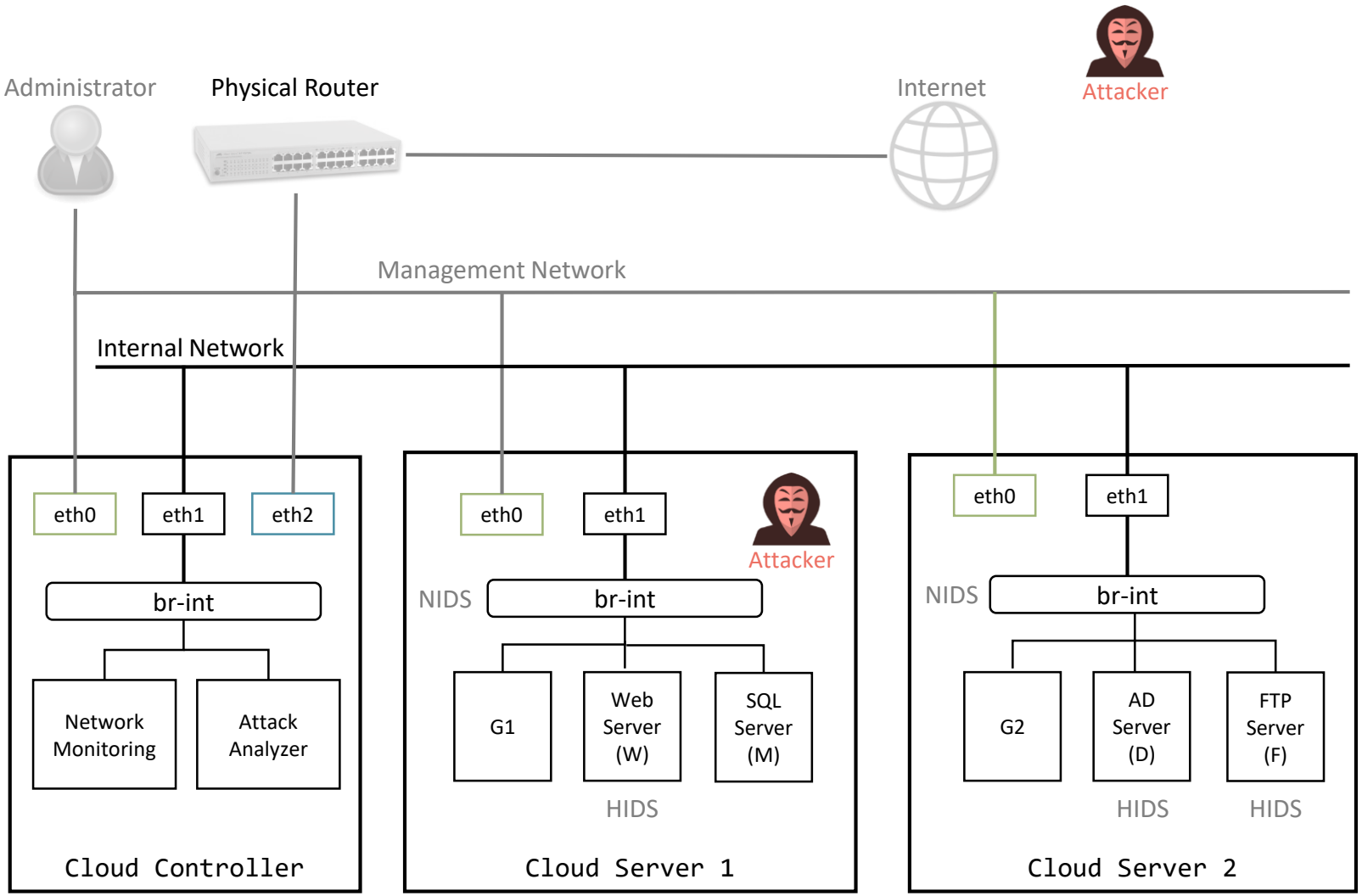
Jha, S et al. 2002, Venkatesan, S et al. 2016

Contents

Intrusion Detection Systems in Cloud Networks

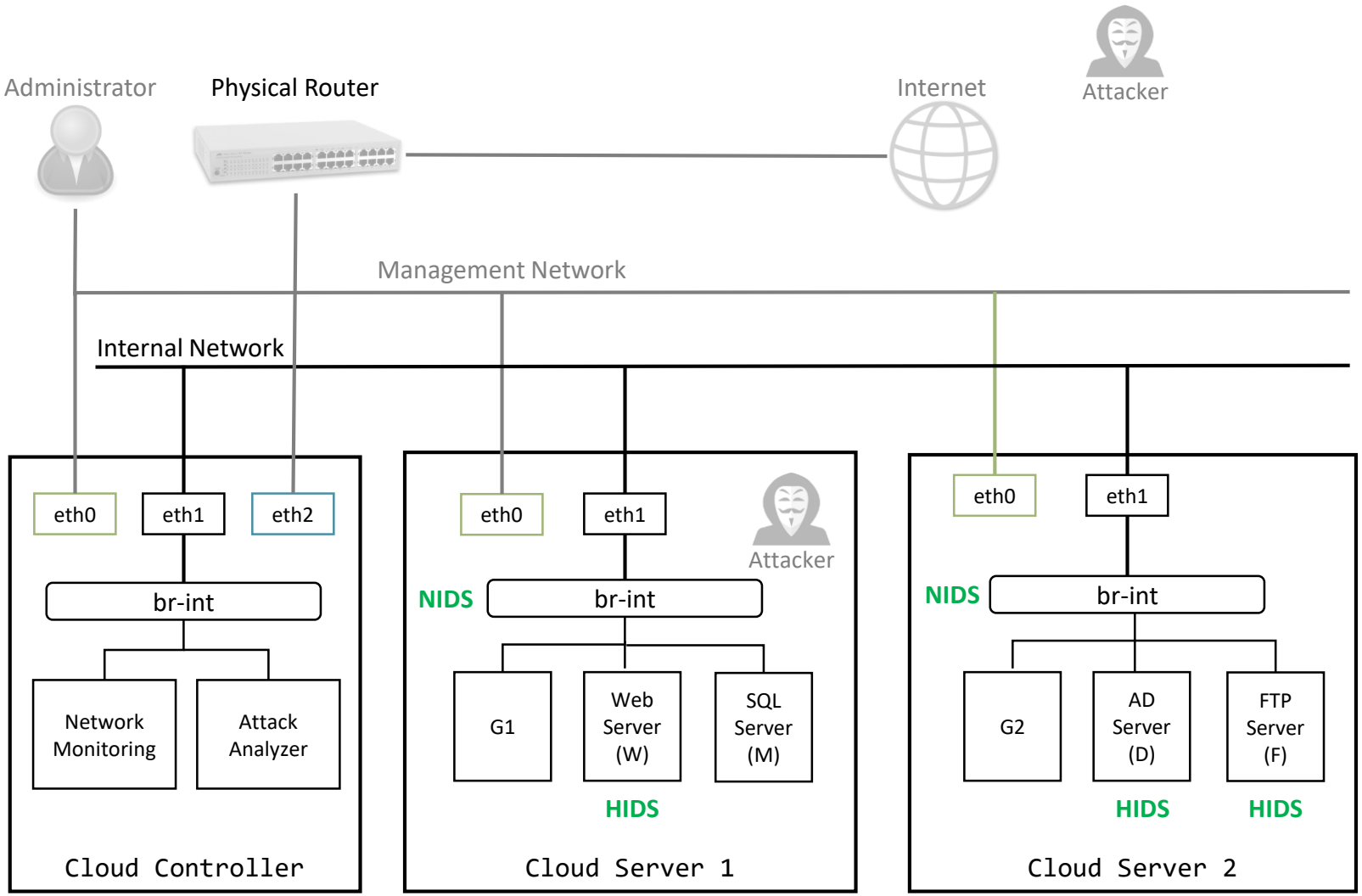


Intrusion Detection Systems in Cloud Networks



Attacker could be located either outside or inside *(stealthy attacker)* the network.

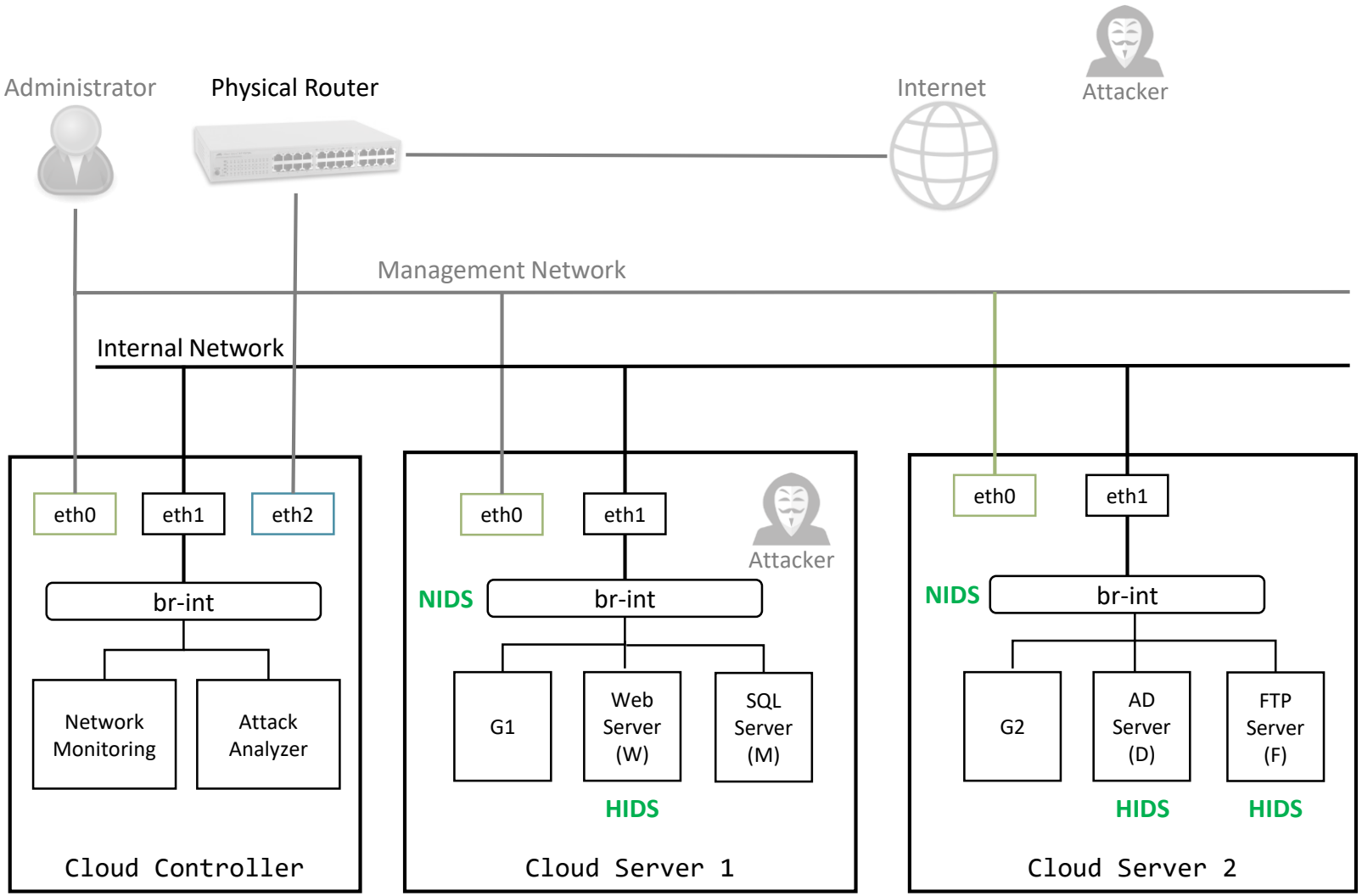
Intrusion Detection Systems in Cloud Networks



Attacker could be located either outside or inside *(stealthy attacker)* the network.

Deploy a limited (k) number of IDS in the Cloud Network (that offer *protection against known vulnerabilities* in the cloud system).

Intrusion Detection Systems in Cloud Networks



Attacker could be located either outside or inside *(stealthy attacker)* the network.

Deploy a limited (k) number of IDS in the Cloud Network (that offer *protection against known vulnerabilities* in the cloud system).

Challenge:
How to place these k Intrusion Detection Systems?

What can we do?

How to place these k Intrusion Detection Systems?

- Static placement of IDS
  - Attacker learns the placement over time and thereby learns how to avoid it.

Moving Target Defense

How to place these k Intrusion Detection Systems?

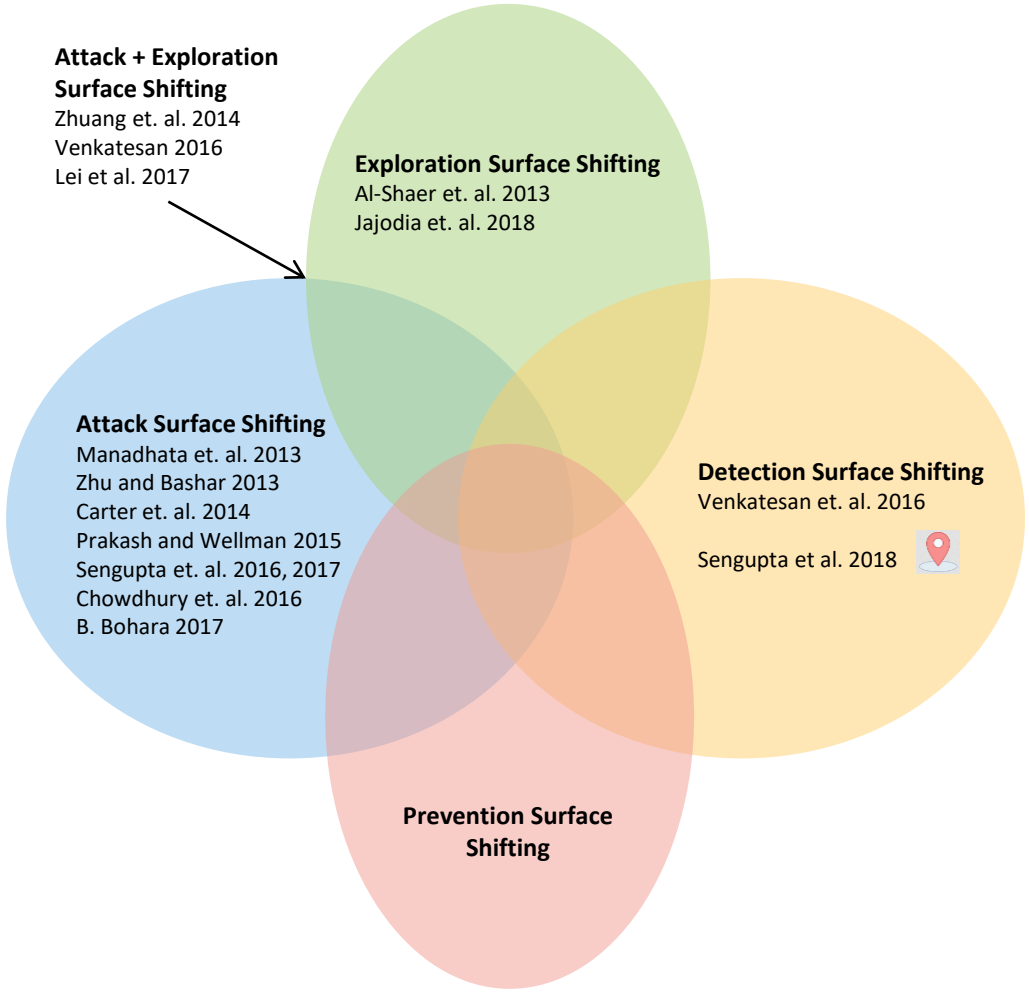
- Static placement of IDS
  - Attacker learns the placement over time and thereby learns how to avoid it.

- Dynamic placement of IDS
  - Keep moving the IDS that are activated at any given point of time

Moving Target Defense

How to place these k Intrusion Detection Systems?

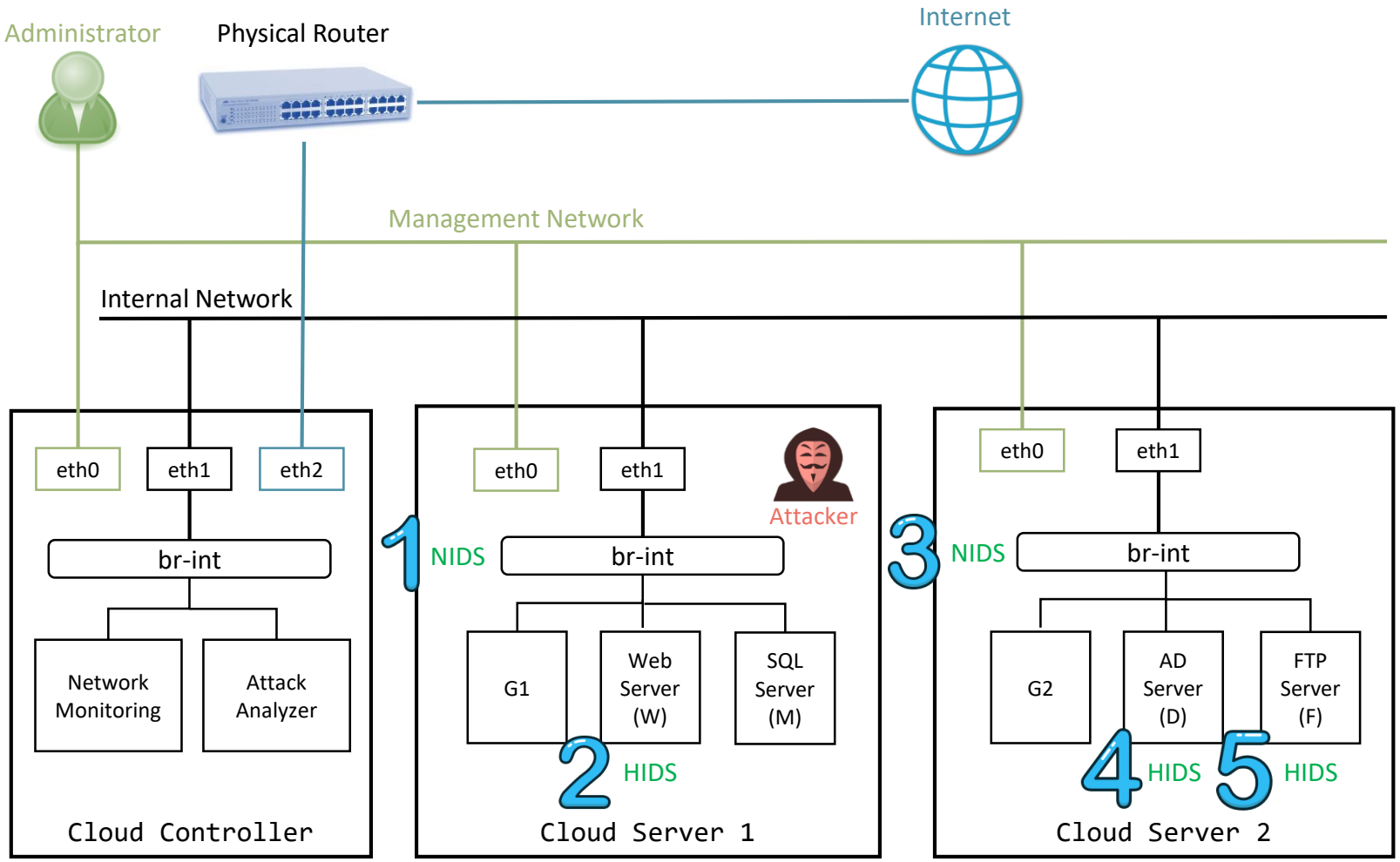
- Dynamic placement of IDS
  - Keep moving the IDS that are activated at any given point of time
  - How to move?
    - Stackelberg Security Game (SSG)



Contents

- Motivations
- Problem Description
- Solution Methods
- Results
- Conclusions

Moving Target Defense – A Cloud Network Scenario



These attacks can be selected from the Common Vulnerabilities and Exposures (CVEs) stored in the National Vulnerability Database (NVD). Each CVE has a

- list of technologies it can effect.
- Expertise required for being able to use it.

1 ⟨192.168.0.6, CVE-2016-0128⟩

2 ⟨192.168.0.6, CVE-2015-1635⟩

3 ⟨192.168.0.6, CVE-2011-0657⟩

4 ⟨192.168.0.7, CVE-2008-5161⟩

5 ⟨192.168.0.9, CVE-2008-5161⟩

Game Theoretic Modeling

Selects a vulnerability to attack

1 2 3 4 5

Number of defender strategies is $\binom{n}{k}$. Combinatorial Explosion!

Selects 2 nodes to deploy IDS in

1 2
1 3
1 4
.
.
.
4 5

Game Theoretic Modeling

Selects a vulnerability to attack

1 2 3 4 5

1

1 2 R^D, R^A

1 3

1 4

. . .

4 5

Selects 2 nodes to deploy IDS in

Number of defender strategies is $\binom{n}{k}$. Combinatorial Explosion!

Thus, the number of utility values that need to be specified is also large!

Efficient Utility Modeling

Number of defender strategies is $\binom{n}{k}$. Combinatorial Explosion!

Thus, the number of utility values that need to be specified is also large!

- ☺ Break it down!
- ☺ Define Utility values for each player for each IDS placement.

Covered	Not covered	Covered	Not covered
1			
$U^D_{c,a}$	$U^D_{u,a}$	$U^A_{c,a}$	$U^A_{u,a}$

Allocated an IDS to detect attack a

Did not.

Common Vulnerability Scoring Service

Common Vulnerability Scoring Systems (CVSS)*

- Is a scoring matrix for CVEs maintained by security experts across the world.

- It has 2 high level scores:
  - Impact Score (IS)
  - Exploitability Score (ES)

- One can generate a Base Score for each CVE based on formulas defined by security experts.

BS = f(IS, ES)

1

Covered Not covered Covered Not covered

$U^{D}_{c,a}$ $U^{D}_{u,a}$ $U^{A}_{c,a}$ $U^{A}_{u,a}$

Obtaining Utility Values

Common Vulnerability Scoring Systems (CVSS)*

- Is a scoring matrix for CVEs maintained by security experts across the world.

- It has 2 high level scores:
  - Impact Score (IS)
  - Exploitability Score (ES)

- One can generate a Base Score for each CVE based on formulas defined by security experts.

BS = f(IS, ES)

	-1 * bet. cen. value	-1*impact	-1 * exp	base
	Covered	Not covered	Covered	Not covered
1	-5.7	-6.4	-8.6	$+6.8$
	$U^D_{c,a}$	$U^D_{u,a}$	$U^A_{c,a}$	$U^A_{u,a}$

Defender's expected utility

$$\max \quad \alpha \cdot \frac{1}{k} \sum_{a \in A} U^{\mathcal{D}}_{c,a} p_a + (1-\alpha) \cdot w_a * U^{\mathcal{D}}_{u,a}(1-p_a)$$

Multi-objective function maximization that,

- Ensures the least impact of performance,
- Maximizes the security

$$s.t. \quad w_a \in \{0,1\} \quad \forall\, a \in A$$

$$p_a \in [0,1] \quad \forall\, a \in A$$

$$p_{t,a} \in [0,1] \quad \forall\, a \in A, t \in T$$

$$\sum_{a \in A} w_a = 1$$

$$\sum_{a \in A} p_{t,a} = 1 \quad \forall\, t \in T$$

$$\sum_{t \in T} p_{t,a} = p_a \quad \forall\, a \in A$$

$$0 \leq v_a - (U^{\mathcal{A}}_{c,a} p_a + U^{\mathcal{A}}_{u,a}(1-p_a)) \leq (1-w_a) * M \quad \forall\, a \in A$$

Defender's expected utility

$$\max \quad \alpha \cdot \frac{1}{k} \sum_{a \in A} U^{\mathcal{D}}_{c,a} p_a + (1-\alpha) \cdot w_a * U^{\mathcal{D}}_{u,a}(1-p_a)$$

$$s.t. \quad w_a \in \{0,1\} \quad \forall\ a \in A$$

$$p_a \in [0,1] \quad \forall\ a \in A$$

$$p_{t,a} \in [0,1] \quad \forall\ a \in A, t \in T$$

$$\sum_{a \in A} w_a = 1$$

$$\sum_{a \in A} p_{t,a} = 1 \quad \forall\ t \in T$$

$$\sum_{t \in T} p_{t,a} = p_a \quad \forall\ a \in A$$

$$0 \leq v_a - (U^{\mathcal{A}}_{c,a} p_a + U^{\mathcal{A}}_{u,a}(1-p_a)) \leq (1-w_a) * M \quad \forall\ a \in A$$

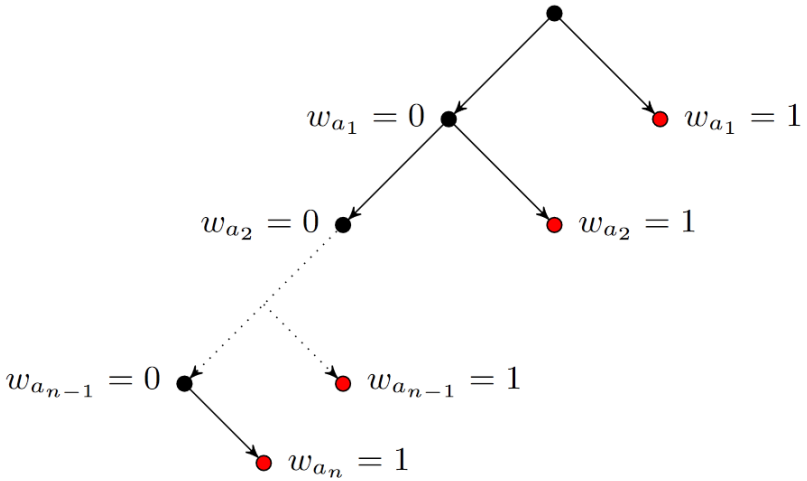
Multi-objective function maximization that,

- Ensures the least impact on performance,
- Maximizes the security

Attacker selects the attack a′ that maximize their utility

$w_{a'} = 1$

Inspired from P Paruchuri et al. 2008



Turns out this is equivalent to solving multiple LPs where you pre-decide the action an attacker will take. Thus, can be computed in polynomial time.

We prove equivalence to a modified version of the multiple LP approach in Korzhyk et al. 2010

Defender's expected utility

$$\max \quad \alpha \cdot \frac{1}{k} \sum_{a \in A} U^{\mathcal{D}}_{c,a} p_a + (1-\alpha) \cdot w_a * U^{\mathcal{D}}_{u,a}(1-p_a)$$

$$s.t. \quad w_a \in \{0,1\} \quad \forall\, a \in A$$

$$p_a \in [0,1] \quad \forall\, a \in A$$

$$p_{t,a} \in [0,1] \quad \forall\, a \in A, t \in T$$

$$\sum_{a \in A} w_a = 1$$

$$\sum_{a \in A} p_{t,a} = 1 \quad \forall\, t \in T$$

$$\sum_{t \in T} p_{t,a} = p_a \quad \forall\, a \in A$$

$$0 \leq v_a - (U^{\mathcal{A}}_{c,a} p_a + U^{\mathcal{A}}_{u,a}(1-p_a)) \leq (1-w_a) * M \quad \forall\, a \in A$$

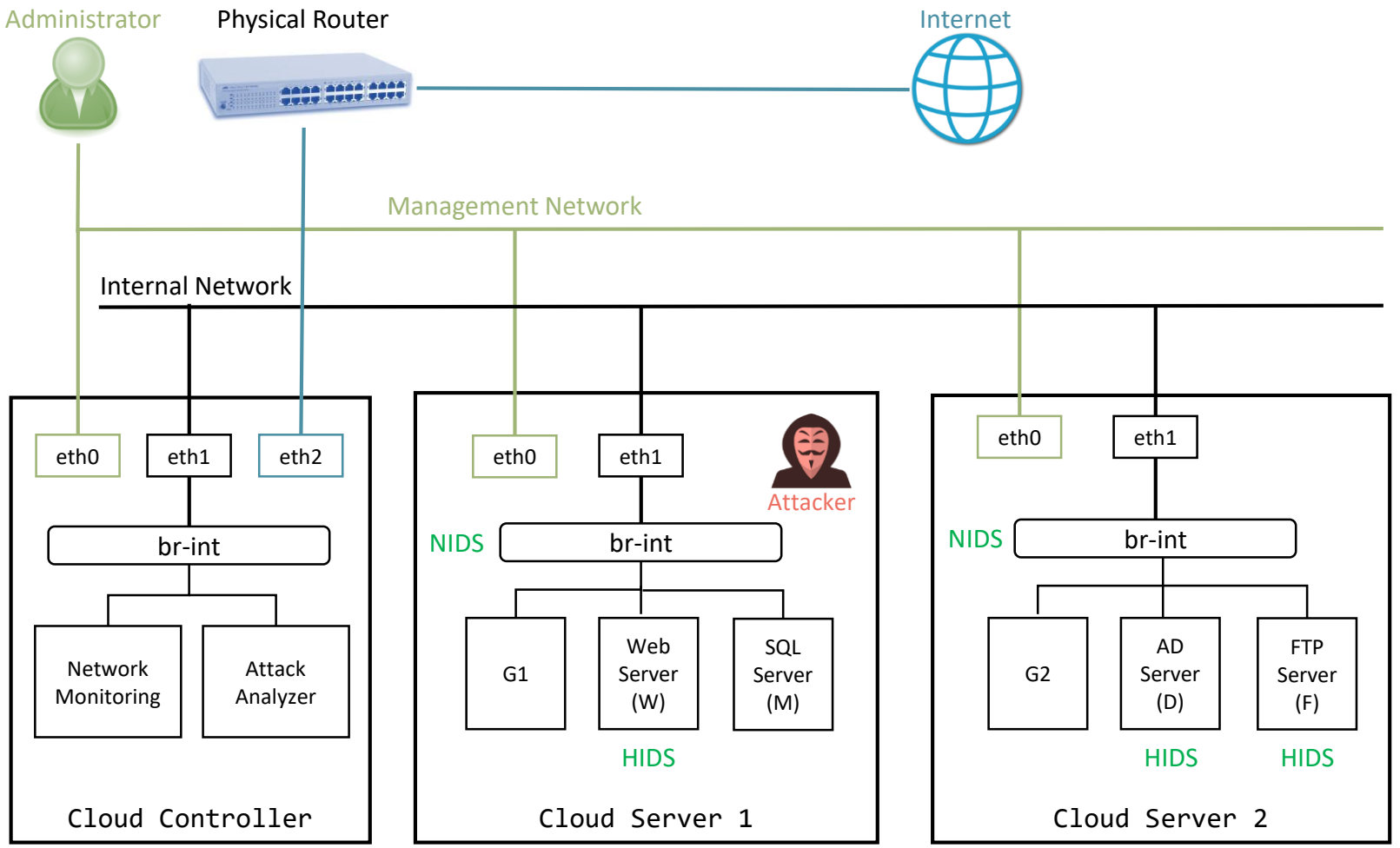
Attacker selects the attack a′ that maximize their utility

$w_{a'} = 1$

Contents

- Motivations
- Problem Description
- Solution Methods
- Results
- Conclusions

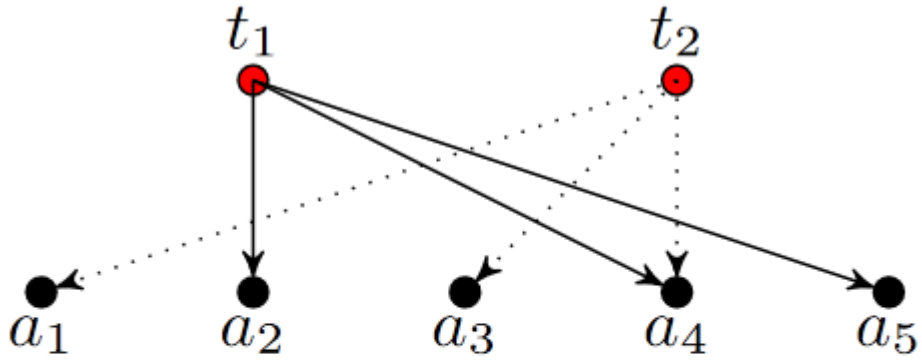
Experiments



Attack	a_1	a_2	a_3	a_4	a_5
$U^{\mathcal{D}}_{c,a}$	-5.7	-10.0	0.0	0.0	0.0
$U^{\mathcal{D}}_{u,a}$	-6.4	-6.4	-2.9	-6.4	-2.9
$U^{\mathcal{A}}_{c,a}$	-8.6	-10	-8.6	-10	-10
$U^{\mathcal{A}}_{u,a}$	6.8	7.5	4.3	7.5	5.0

Table 2. Player utilities for each vulnerability depending on whether (or not) an IDS is deployed to detect the attacks that exploit it.

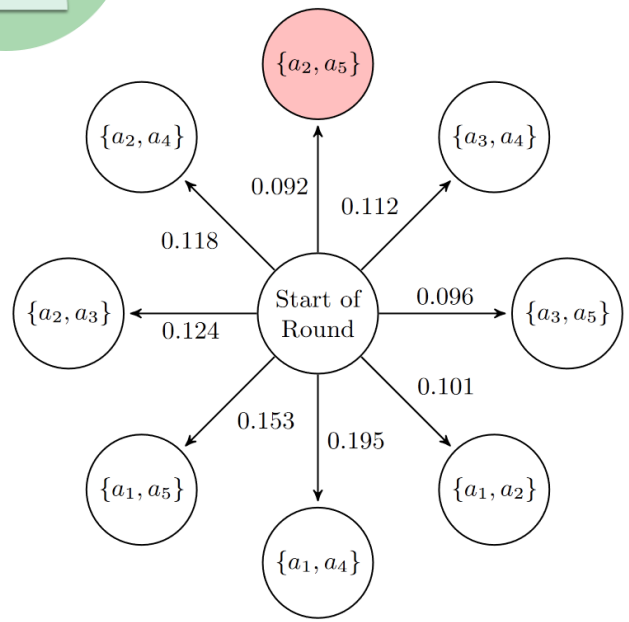
Finding implementable strategies

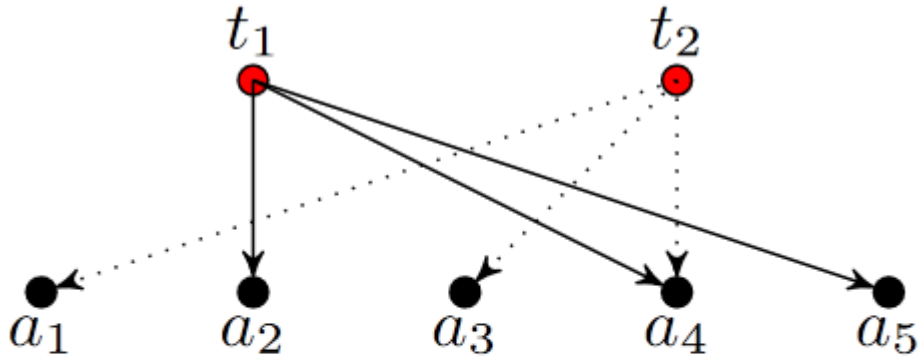


	a_1	a_2	a_3	a_4	a_5
t_1	0	0.44	0	0.22	0.34
t_2	0.45	0	0.34	0.21	0

$p_{t,a}$

Finding implementable strategies



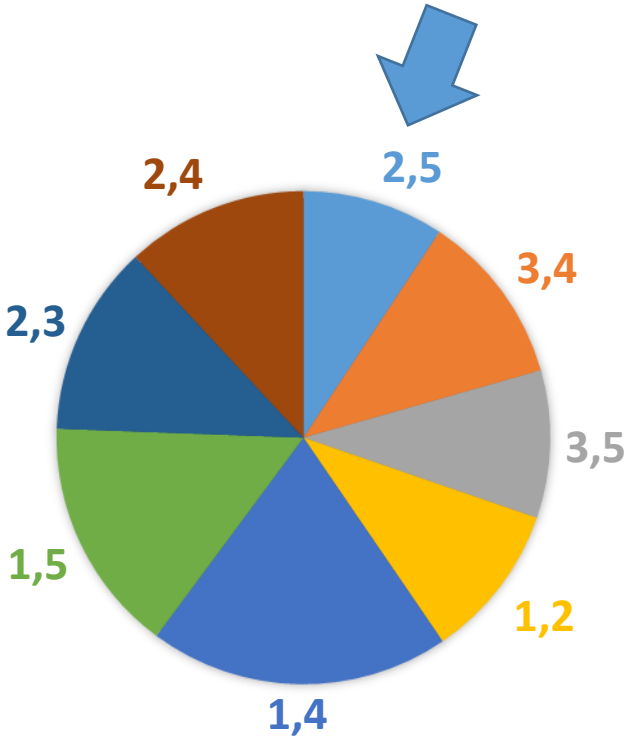


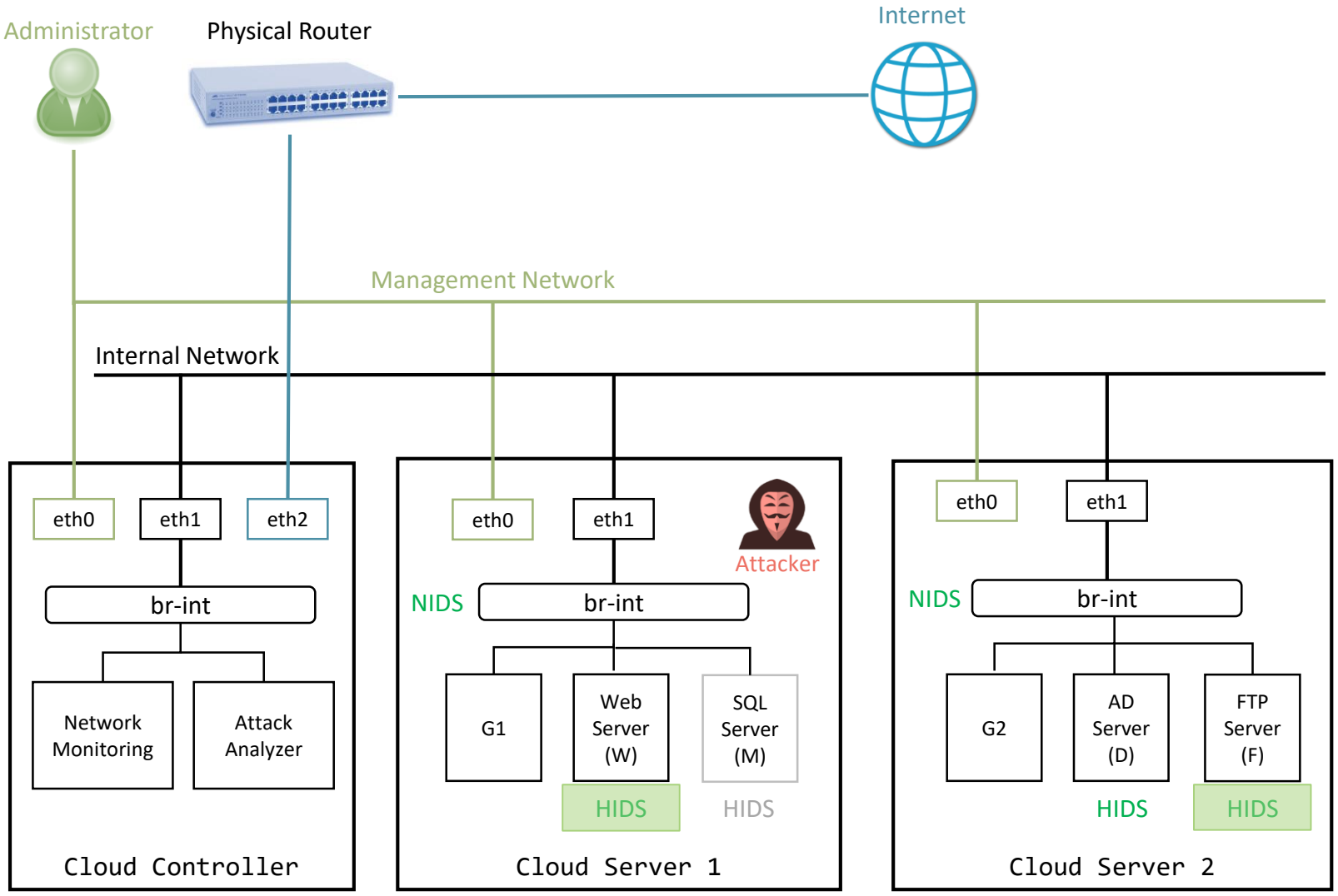
	a_1	a_2	a_3	a_4	a_5
t_1	0	0.44	0	0.22	0.34
t_2	0.45	0	0.34	0.21	0

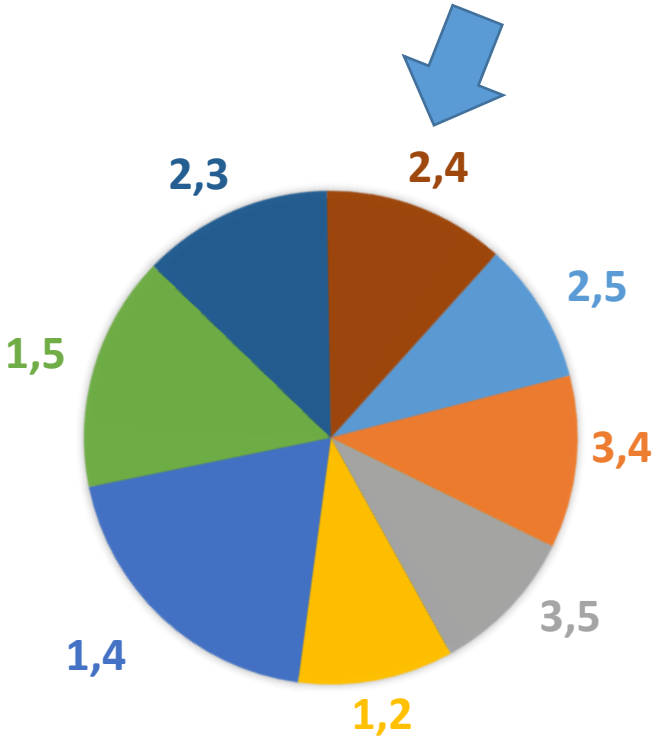
$p_{t,a}$

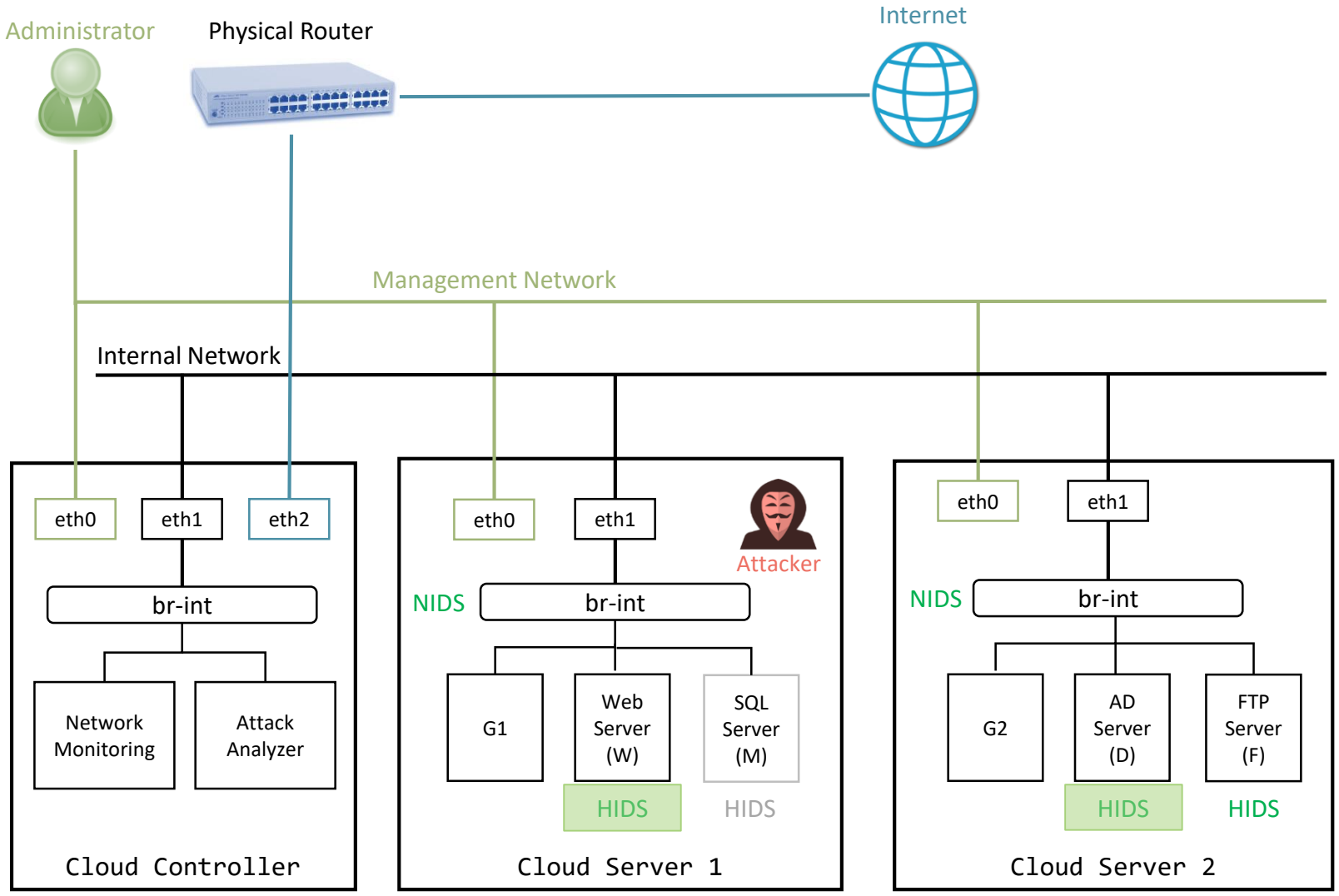
Birkhoff Von-Neumann Theorem

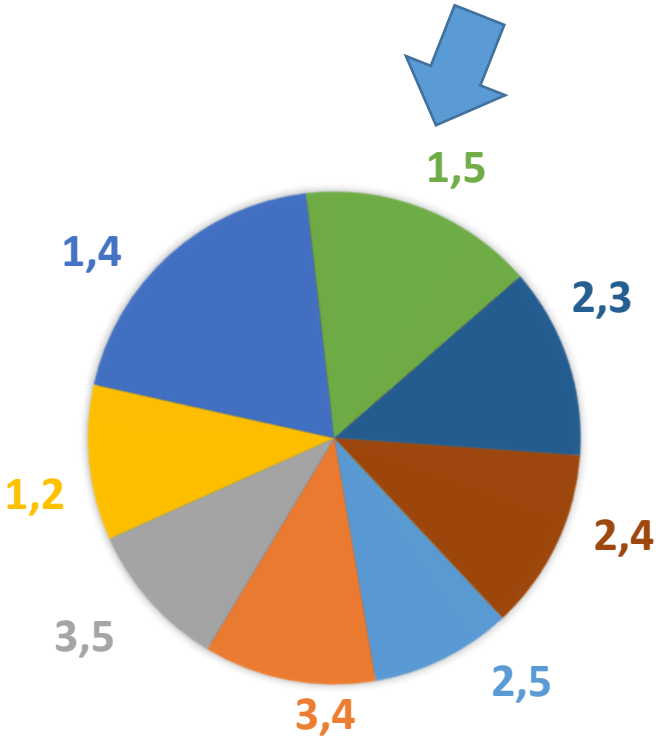
Used implementation from Budish, Eric, et al. "Designing random allocation mechanisms: Theory and applications." *American Economic Review* 103.2 (2013): 585-623.

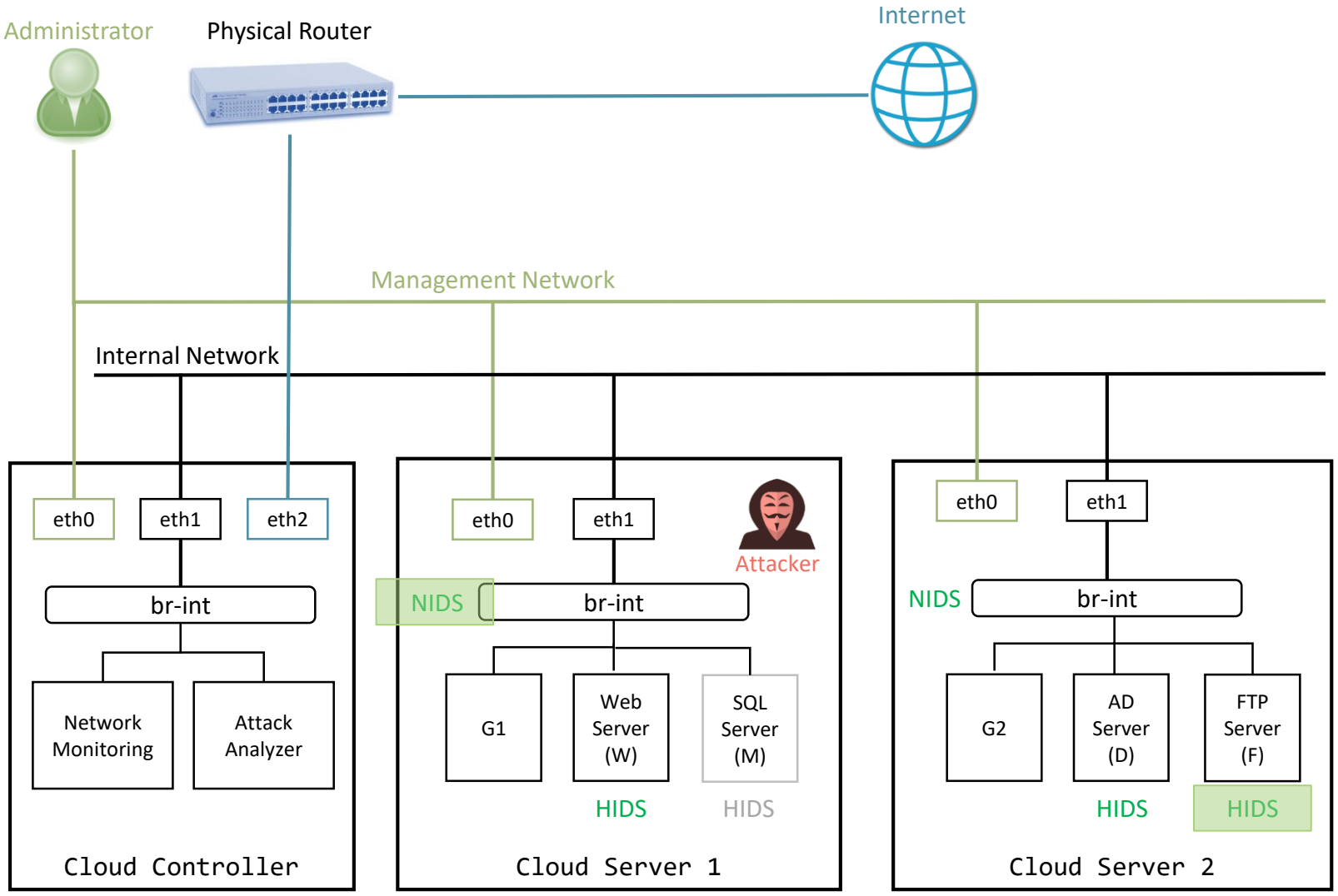

2,5
2,4
3,4
2,3
3,5
1,5
1,2
1,4


Administrator
Physical Router
Internet
Management Network
Internal Network
eth0
eth1
eth2
br-int
Network Monitoring
Attack Analyzer
Cloud Controller
eth0
eth1
Attacker
NIDS
br-int
G1
Web Server (W)
SQL Server (M)
HIDS
HIDS
Cloud Server 1
eth0
eth1
NIDS
br-int
G2
AD Server (D)
FTP Server (F)
HIDS
HIDS
Cloud Server 2


2,3
2,4
2,5
1,5
3,4
3,5
1,4
1,2


Administrator
Physical Router
Internet
Management Network
Internal Network
eth0
eth1
eth2
br-int
Network Monitoring
Attack Analyzer
Cloud Controller
eth0
eth1
Attacker
NIDS
br-int
G1
Web Server (W)
SQL Server (M)
HIDS
HIDS
Cloud Server 1
eth0
eth1
NIDS
br-int
G2
AD Server (D)
FTP Server (F)
HIDS
HIDS
Cloud Server 2


1,5
1,4
2,3
1,2
2,4
3,5
2,5
3,4


Administrator
Physical Router
Internet
Management Network
Internal Network
eth0
eth1
eth2
br-int
Network Monitoring
Attack Analyzer
Cloud Controller
eth0
eth1
Attacker
NIDS
br-int
G1
Web Server (W)
SQL Server (M)
HIDS
HIDS
Cloud Server 1
eth0
eth1
NIDS
br-int
G2
AD Server (D)
FTP Server (F)
HIDS
HIDS
Cloud Server 2

Comparison to state-of-the-art mechanisms

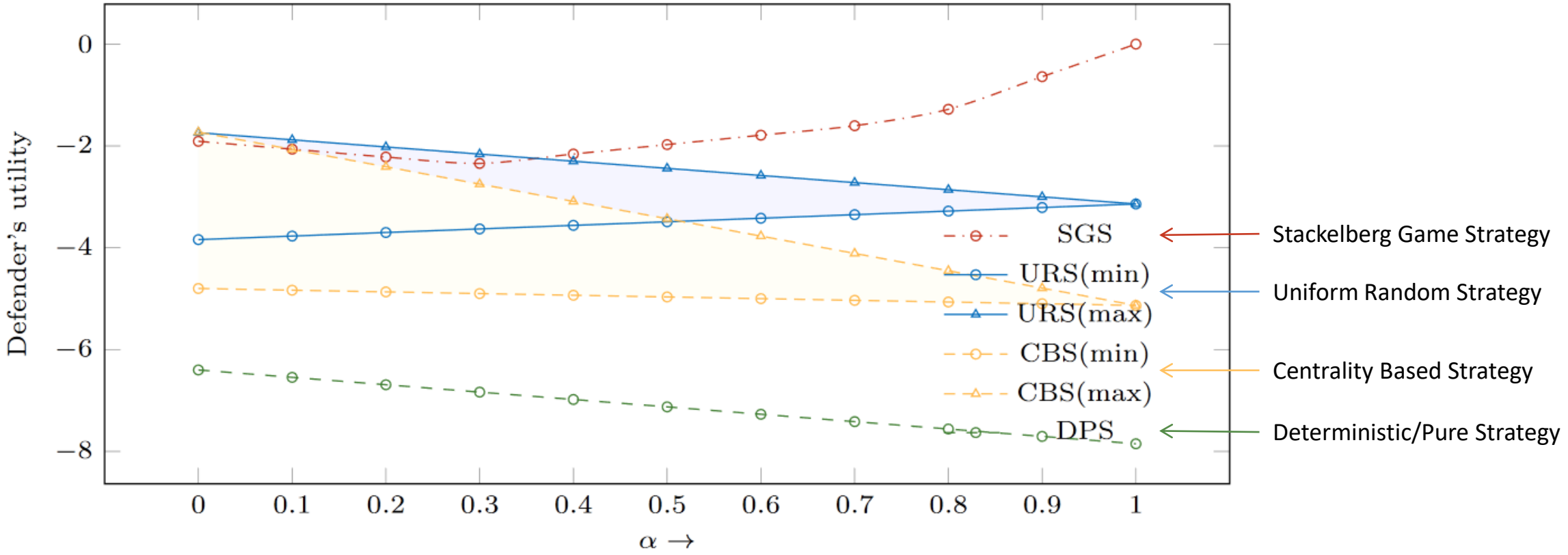


Fig. 5. Defender's utility for the various MTD strategies as the security-usability trade-off value (α) varies from zero to one.

Finding the Most Critical Vulnerabilty

- The question of removing the most critical vulnerability now has to reason about the multi objective function.

Eg. a high impact vulnerability which does not effect the performance could always be covered and thus a vulnerability with lesser impact should be fixed first.

- We suggest a brute force algorithm that removes the vulnerability that yield the maximum gain in defender utility.

Question: Is there a sub-problem structure that can be exploited here to use the solution for the most critical vulnerability to find the k critical vulnerabilities?

How many IDS to deploy?

We use this method on a cloud system with 15 VM network that has 42 vulnerabilities distributed among them.

Even when weightage on performance is low, we notice that, going beyond 30 IDS makes the performance cost outweigh the security benefits.

Can be seen as a precomputation step.

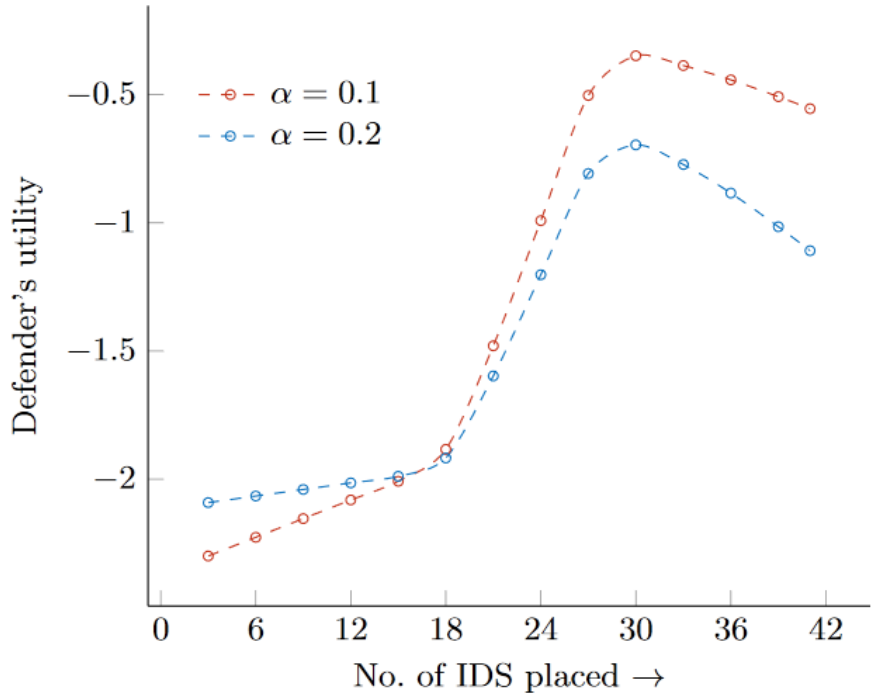


Fig. 8. Change in defender's utility value as the number of NIDS and HIDS deployed increases.

Contents

- Motivations
- Problem Description
- Solution Methods
- Results
- Conclusions

Conclusion

Showed that using more NIDS and HIDS systems in a cloud network setting impacts performance, thus motivating the need for limited use of NIDS and HIDS placement.

Introduced the concept of Moving Target Defense (MTD) for dynamic placement of Intrusion Detection Systems (IDS) systems.

Formulated it as a Stackelberg Security Game (SSG) and designed a polynomial time solver to calculate the marginal probabilities of deploying IDS against a particular attack.

Showed how the effectiveness of the mixed strategy in comparison to state-of-the art in the cybersecurity domain.

Discussed selection of the number of resources for an actual cloud system.

Introduced and proposed a brute force solution to the problem of finding the most critical vulnerability.