65th Meeting of IFIP Working Group 10.4 on
Dependable Computing and Fault Tolerance

Toward Scalable Security Models and Analysis

Dong-Seong (Dan) Kim

University of Canterbury

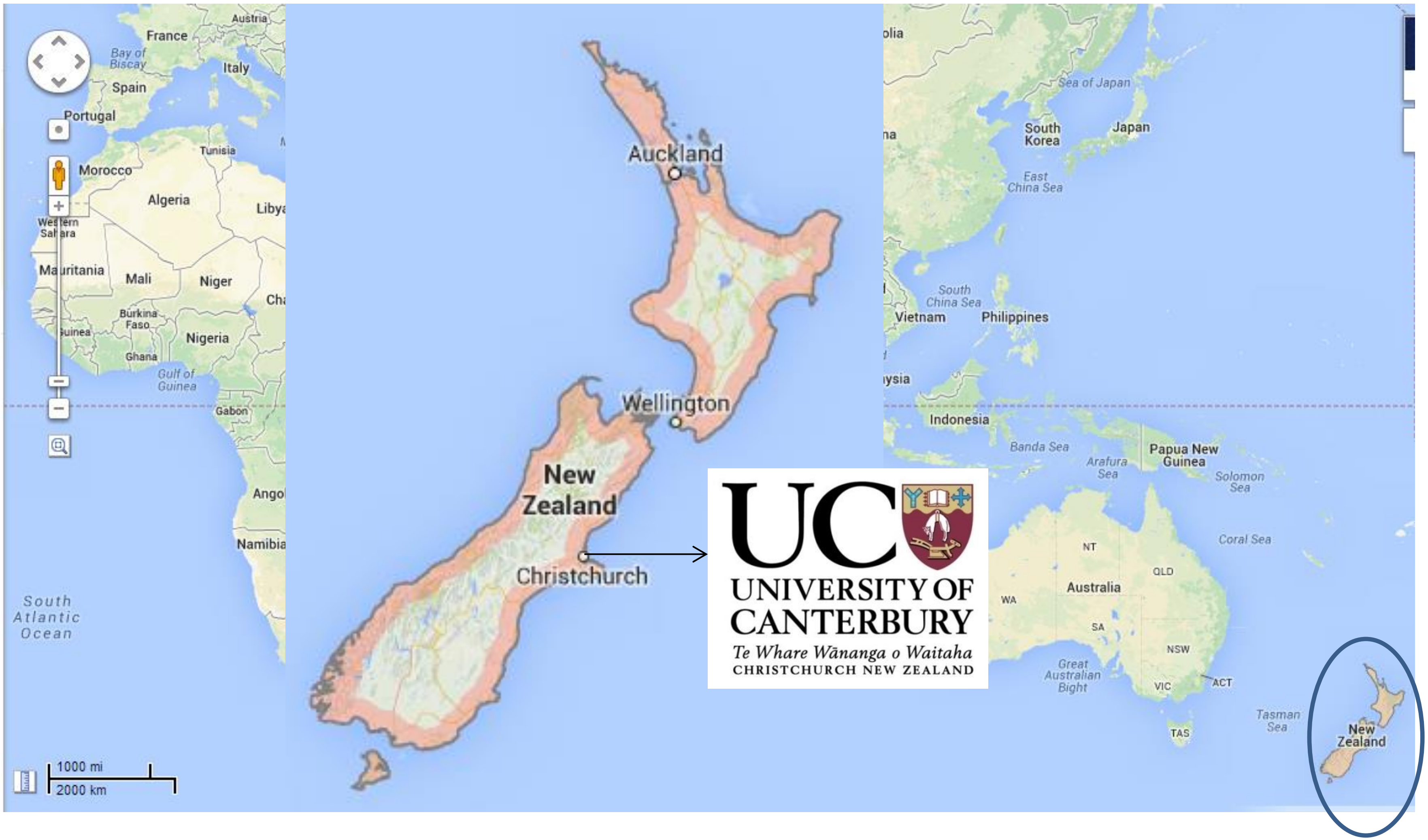
Christchurch, New Zealand

Email: dongseong.kim@canterbury.ac.nz

Outline

- Introduction
- Problems
- Our proposed ideas
  - HARMs
  - Simplified HARM in construction
  - Simplified HARM in evaluation
- Summary

New Zealand



France
Austria
Bay of Biscay
Italy
Spain
Portugal
Tunisia
Morocco
Algeria
Libya
Western Sahara
Mauritania
Mali
Niger
Burkina Faso
Nigeria
Ghana
Gulf of Guinea
Gabon
Angol
Namibia
South Atlantic Ocean
1000 mi
2000 km
Auckland
Wellington
New Zealand
Christchurch
UC
UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND
Sea of Japan
South Korea
Japan
East China Sea
South China Sea
Vietnam
Philippines
Indonesia
Banda Sea
Arafura Sea
Papua New Guinea
Solomon Sea
Coral Sea
NT
QLD
Australia
WA
SA
NSW
Great Australian Bight
VIC
ACT
TAS
Tasman Sea
New Zealand

NATO Emerging Security Challenges Division Science for Peace and Security Programme (SPS)

NATO Country (USA)

NATO Partner Country (Morocco)

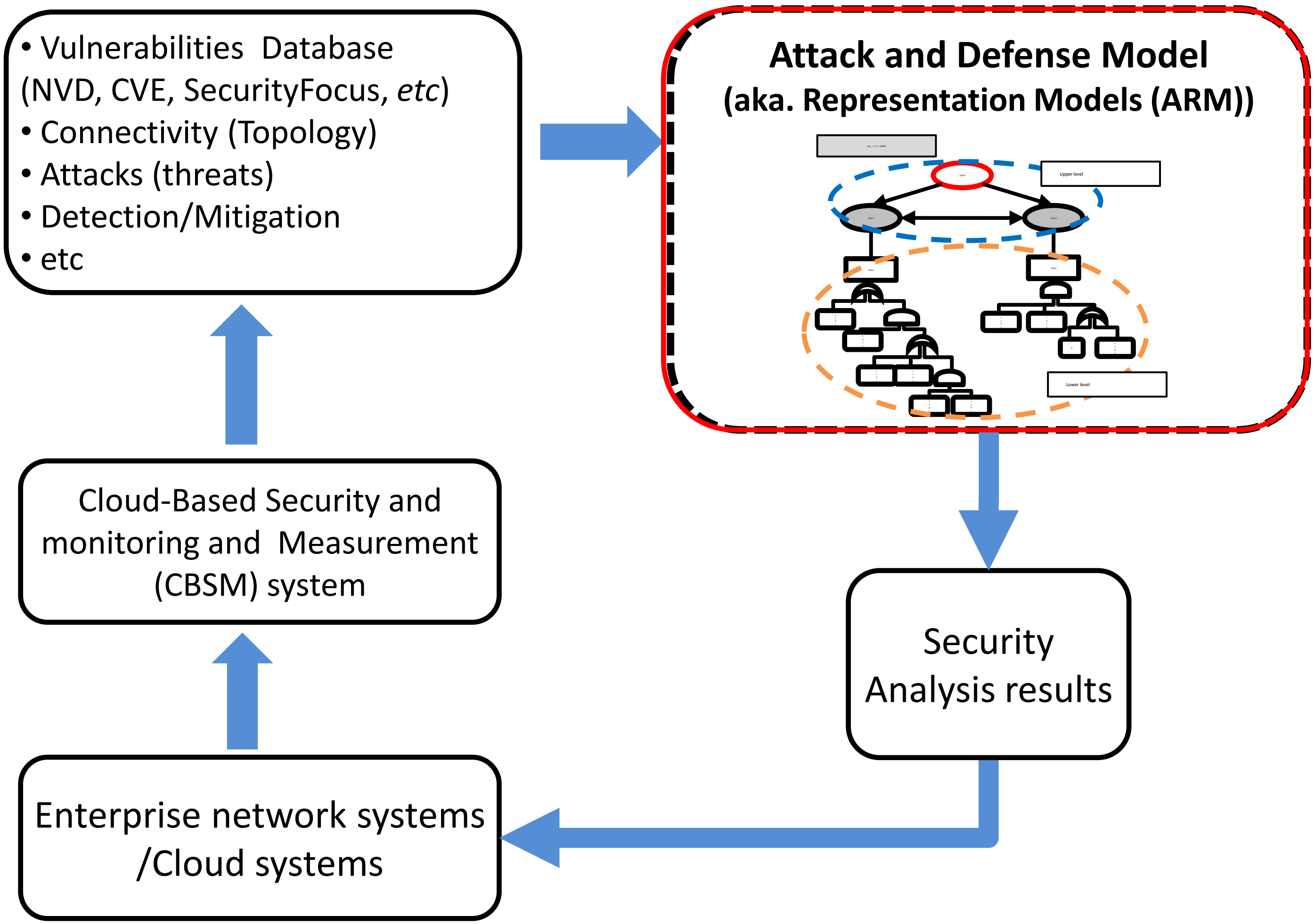
NPD: Prof. K. Trivedi, Duke Uni.
Co-Director: Assoc. Prof. D. Huang, ASU

PPD: Prof. A. Haqiq,
Hassan 1st University

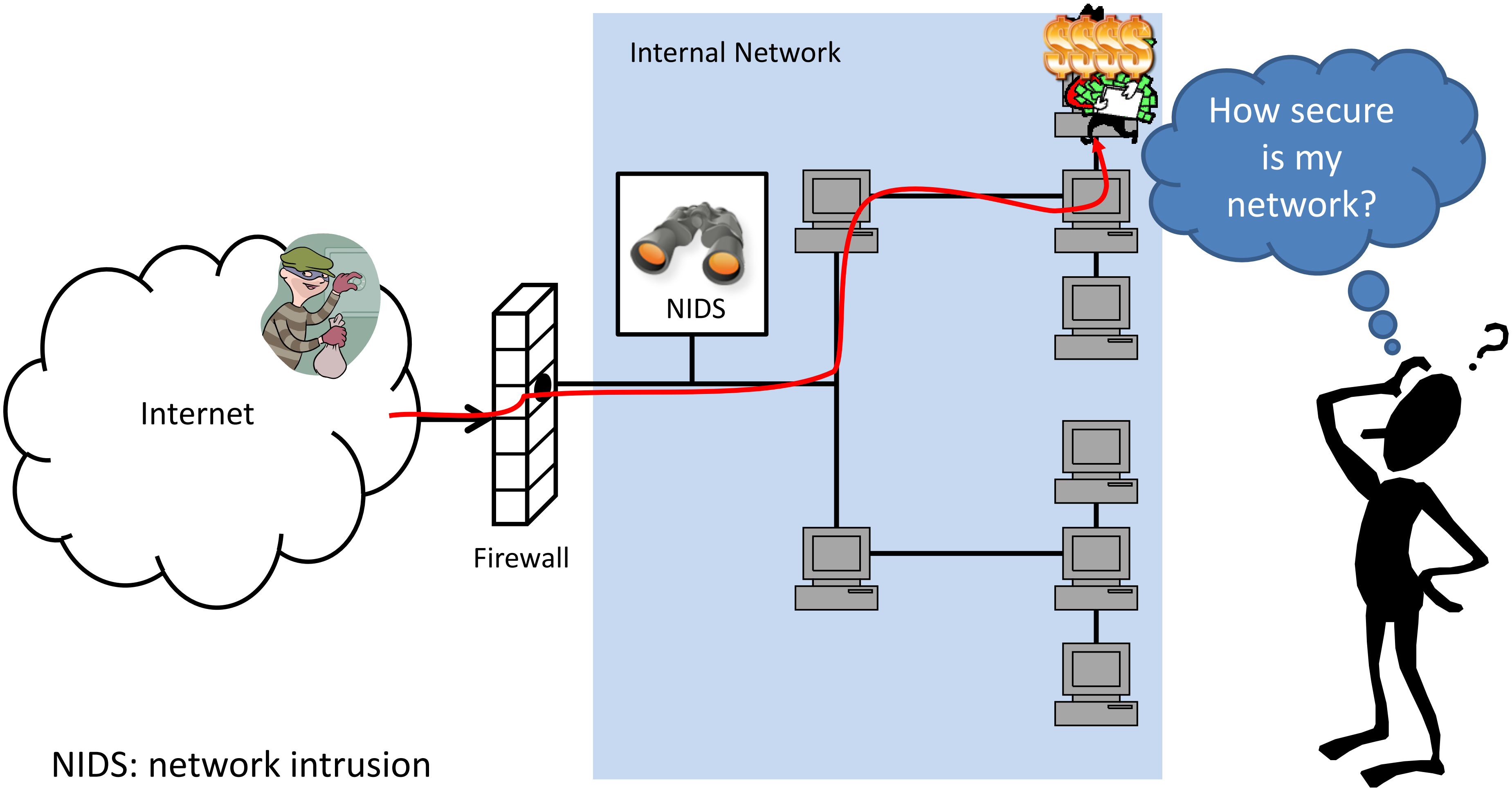
Major non-NATO allies Country

Dr. Dong-Seong Kim
(U of Canterbury, New Zealand)

Cyber Security Analysis and Assurance using Cloud-Based Security Measurement System



Security Assessment

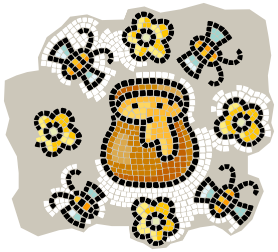


NIDS: network intrusion detection system

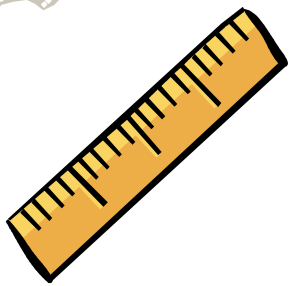
How to assess security?

Security Assessment (cont.)

- To assess security, one requires **3M**s:



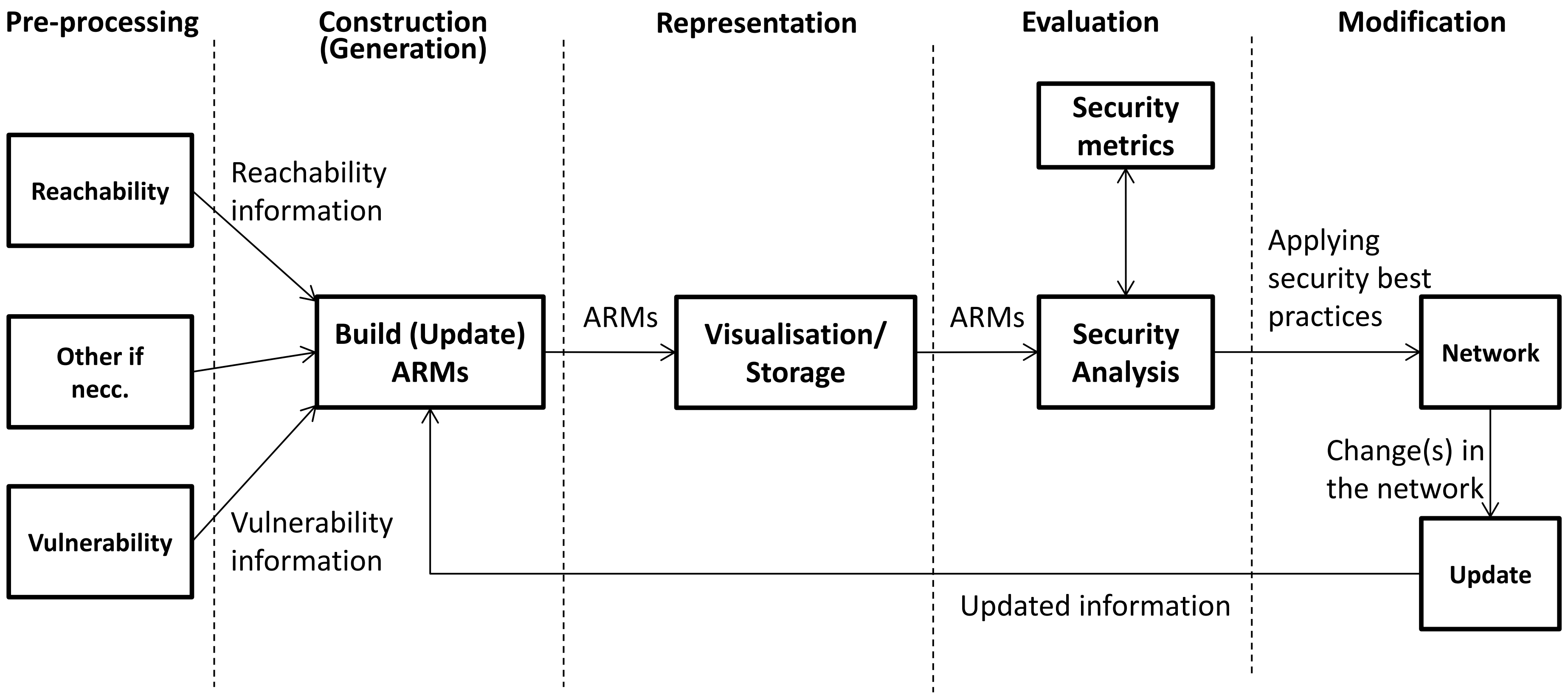
1. Security **M**easures
   - To **collected** required information.



2. Security **M**etrics
   - To **represent** the analysis of security.
3. Security **M**odels (Attack Representation Model: ARM)
   - To **capture** security using simulation, analytic models, or hybrid models.

Lifecycle of ARM?

Attack Representation Model (ARM)* life cycles

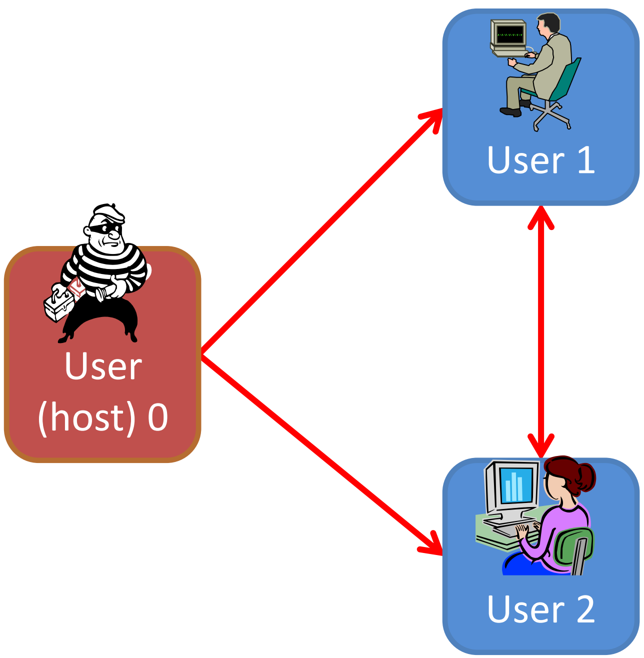


*aka., Attack and Defense Models

*an example?

An example network and AG

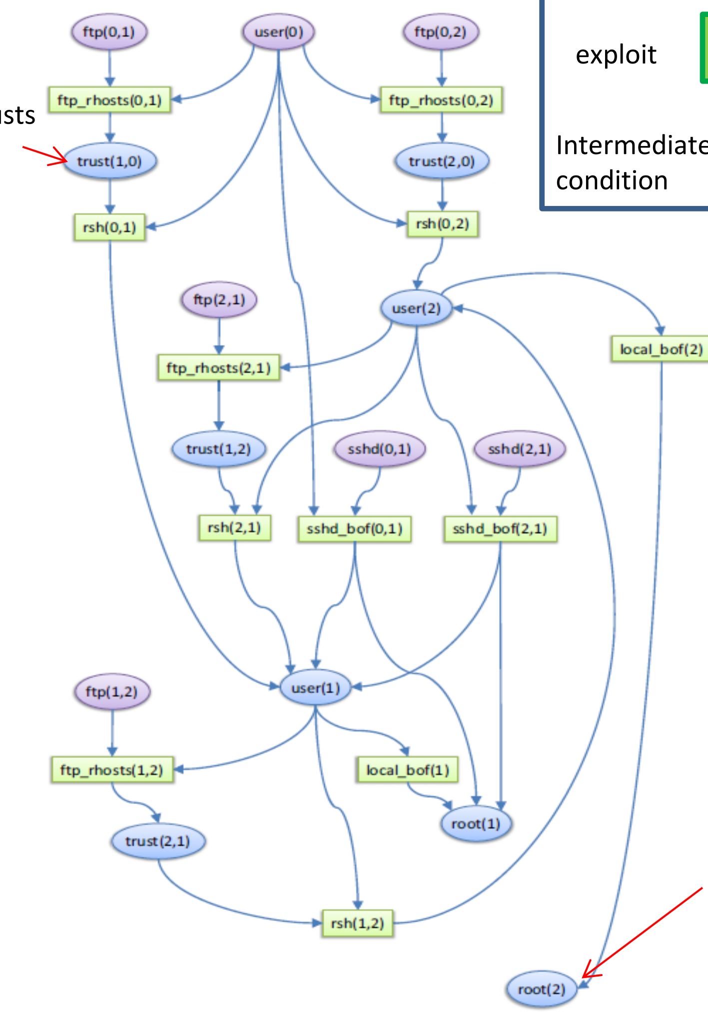
Security objective: to harden the network w.r.t target condition root(2)



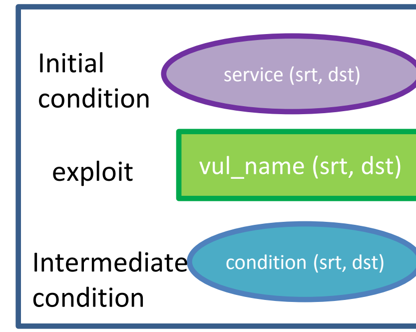
Vulnerabilities:
- ftp_rhosts
- rsh
- sshd_bof
- local_bof

Vulnerabilities:
- ftp_rhosts
- rsh
- local_bof

Attack graph (AG)



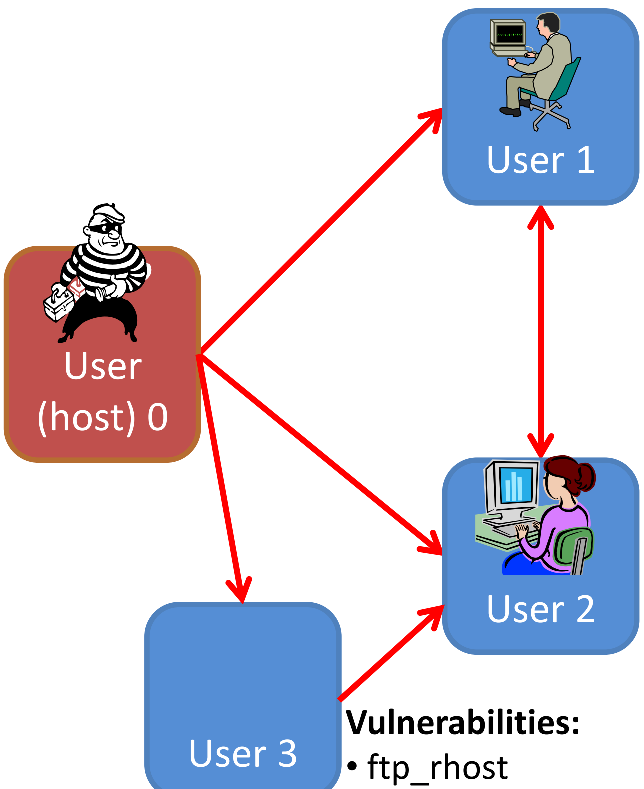
Legend



M. Albanese, S.Jajodia, S. Noel, "**A Time-Efficient and Cost Effective Network Hardening Using Attack Graphs**", in Proc. IEEE DSN 2012

An example network and AG

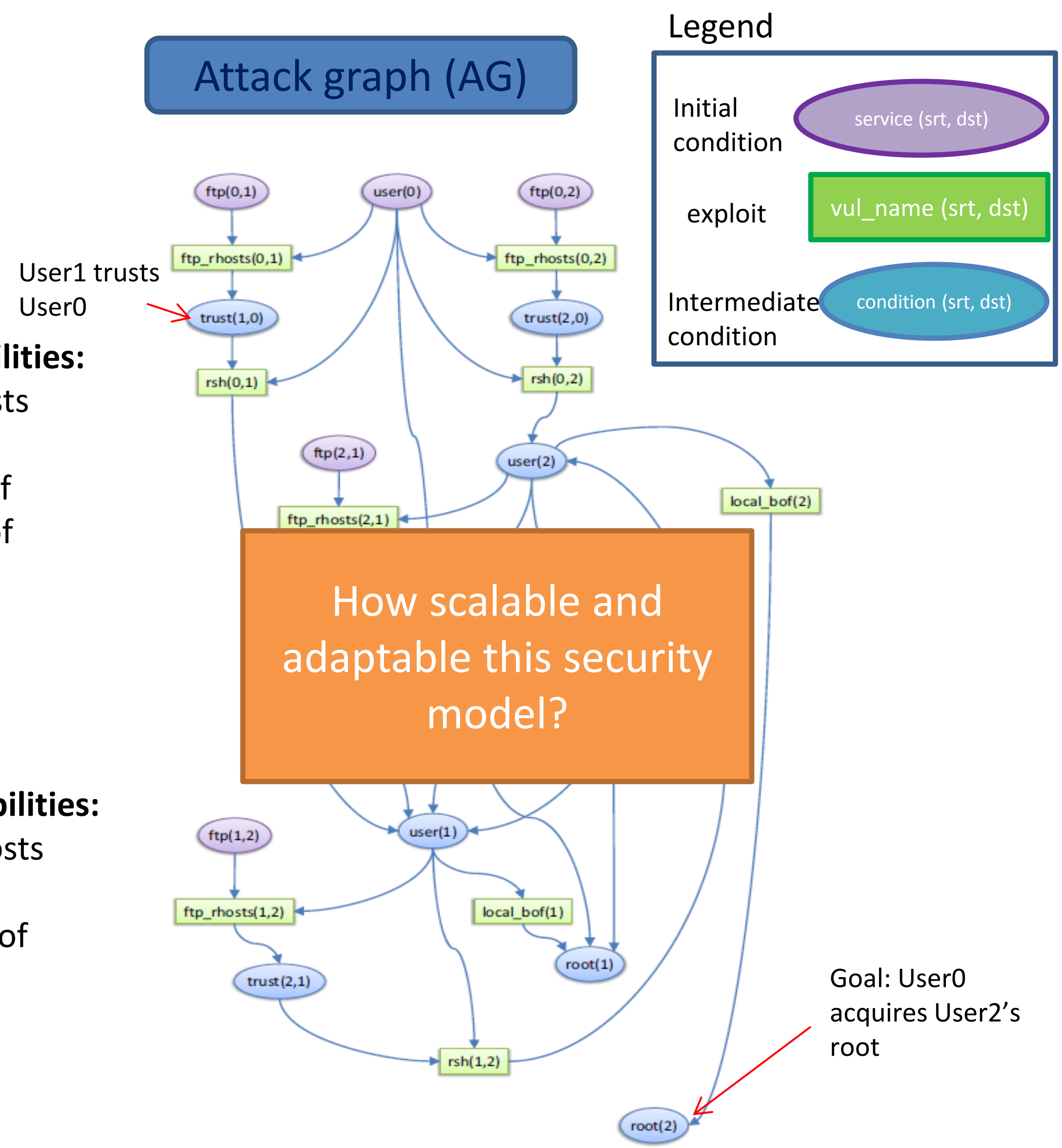
Security objective: to harden the network w.r.t target condition root(2)



Vulnerabilities:
- ftp_rhosts
- rsh
- sshd_bof
- local_bof
- web_bof

Vulnerabilities:
- ftp_rhosts
- rsh
- local_bof

Vulnerabilities:
- ftp_rhost
- local_bof



M. Albanese, S.Jajodia, S. Noel, "**A Time-Efficient and Cost Effective Network Hardening Using Attack Graphs**", in Proc. IEEE DSN 2012

Two issues on ARMs

- **Scalability issues**
  - The generation/evaluation of full attack models (all possible attack scenarios) exhibit a **state-space explosion.**

- **Dynamic adjustment issues**
  - A change in the network system causes **reconstruction** (in worst case) of the ARMs.

R. E. Sawilla and X. Ou, "Identifying critical attack assets in dependency attack graphs," in Proc. ESORICS2008, pp. 18–34.

Dealing with Scalability

1. Using Hierarchical ARMs (HARMs)
   – Modelling hosts and vulnerabilities in two different layers (i.e., 2-level hierarchy).
   – Simulation result

2. Construct ARMs based on Important components
   – Improve the construction complexity using less components.

3. Security Analysis based on Important components
   – Using important hosts and vulnerabilities for security analysis.

Our proposed idea

Use of two-level *Hierarchical* ARMs (HARMs)

Buda (higher area)

Budapest= Danube river

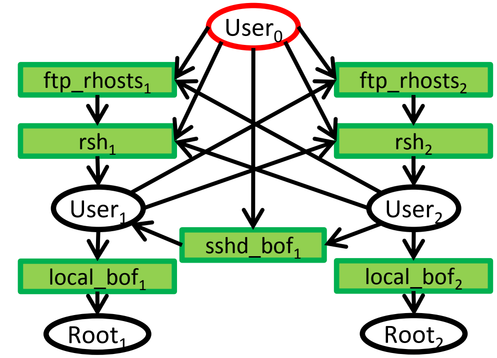
+pest (lower area)

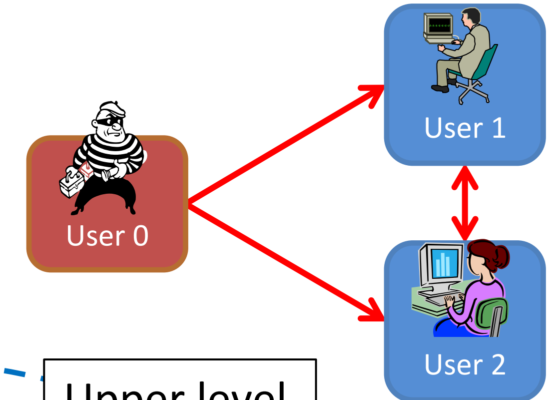
Represent the *network path information* in the upper level and *vulnerability exploitation information* in the lower level

Note that this can be extended to multi-level Hierarchical model.

J. Hong, D. Kim,"HARMs: Hierarchical Attack Representation Models for Network Security Analysis" in Proc. of the 10th Australian Information Security Management Conference (SECAU 2012)

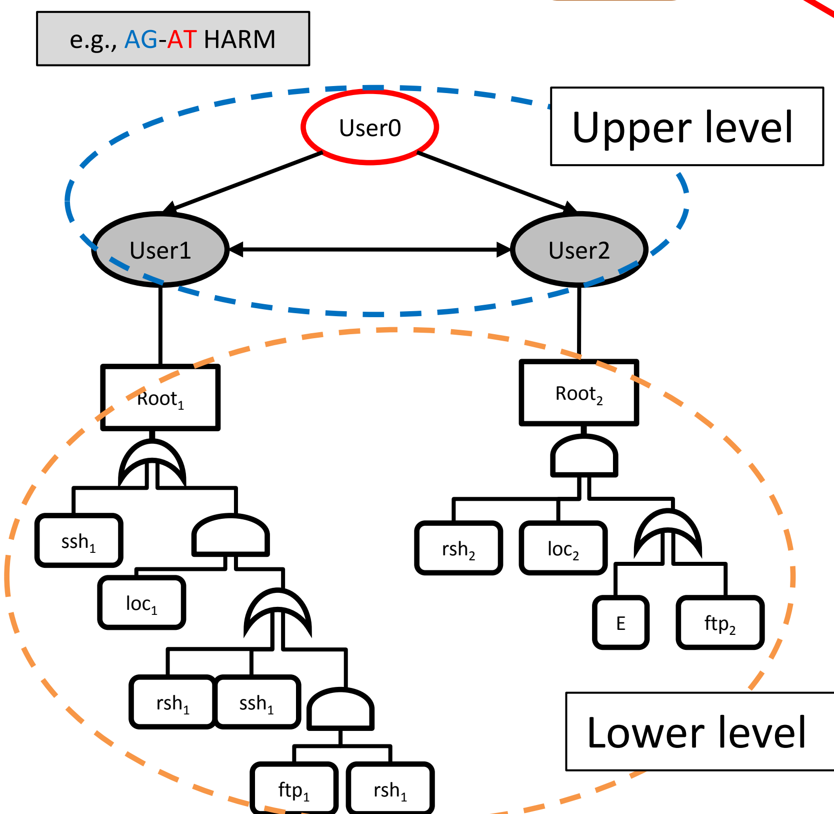
Example of HARMs

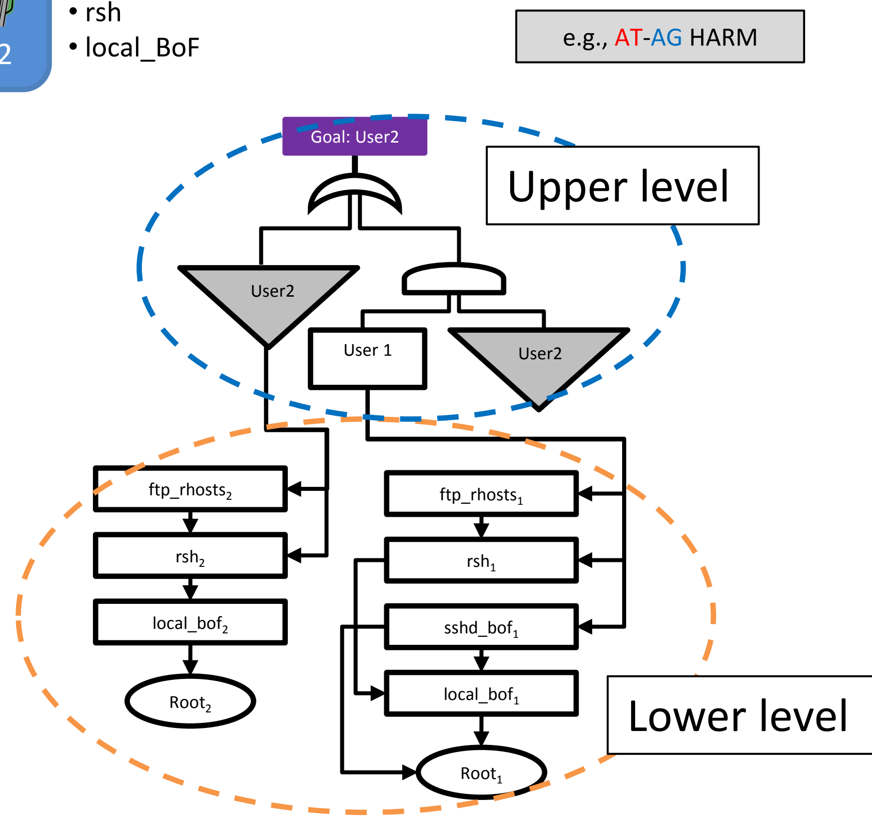




Vulnerabilities:
- ftp_rhosts
- rsh
- sshd_BoF
- local_BoF

Vulnerabilities:
- ftp_rhosts
- rsh
- local_BoF



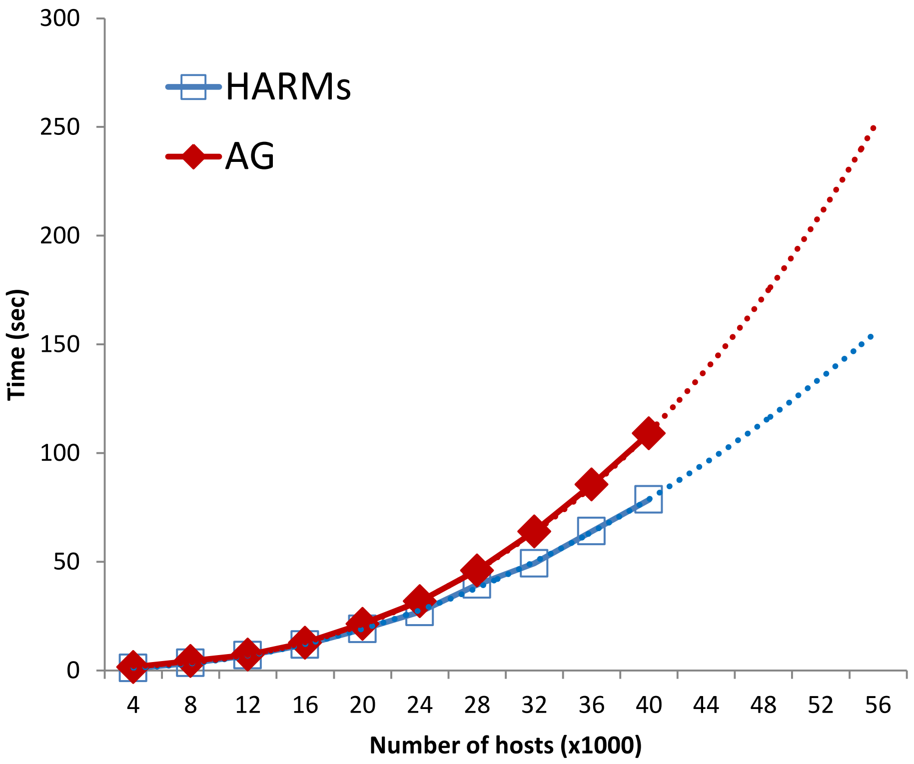


Performance Evaluation via Simulation

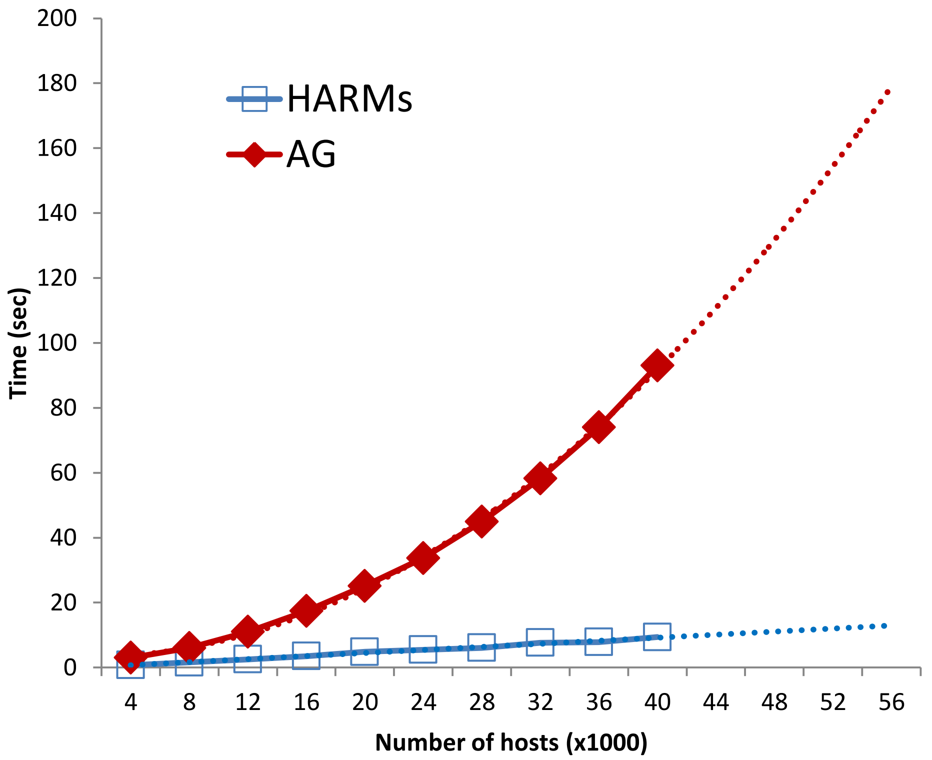
- Assume an external attacker and a target
- Consider
  - **performances** in construction and evaluation phase for an AG and an HARM (AG-AG type)
    - Time, number of computations
  - **various** network topologies
    - Fully connected, ring and star
  - **variable** number of vulnerabilities
    - Hosts are assigned with varying number of vulnerabilities
  - **different** vulnerability types
    - Vulnerabilities to gain different level of privileges (e.g., user/root)

Performance Evaluation via Simulations (cont.)

- Simulation 1 – fully connected topology, bounded attack path length



Construction



Evaluation

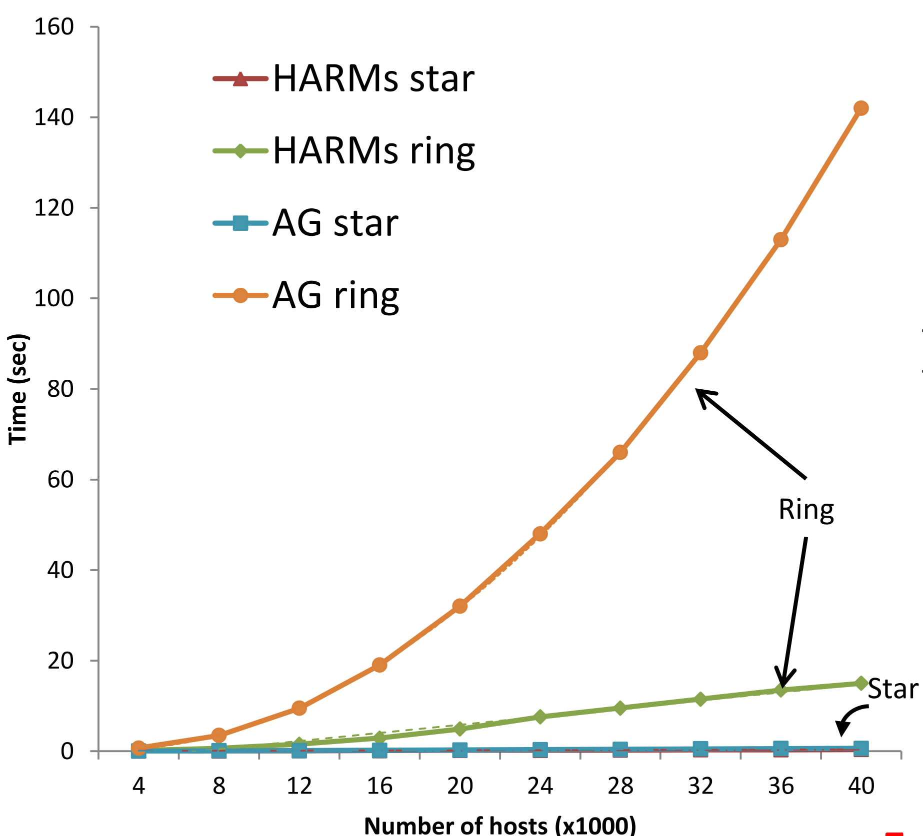
Fixed No. of Vulnerabilities = 10
(1 remote-to-root, 9 remote-to-other)

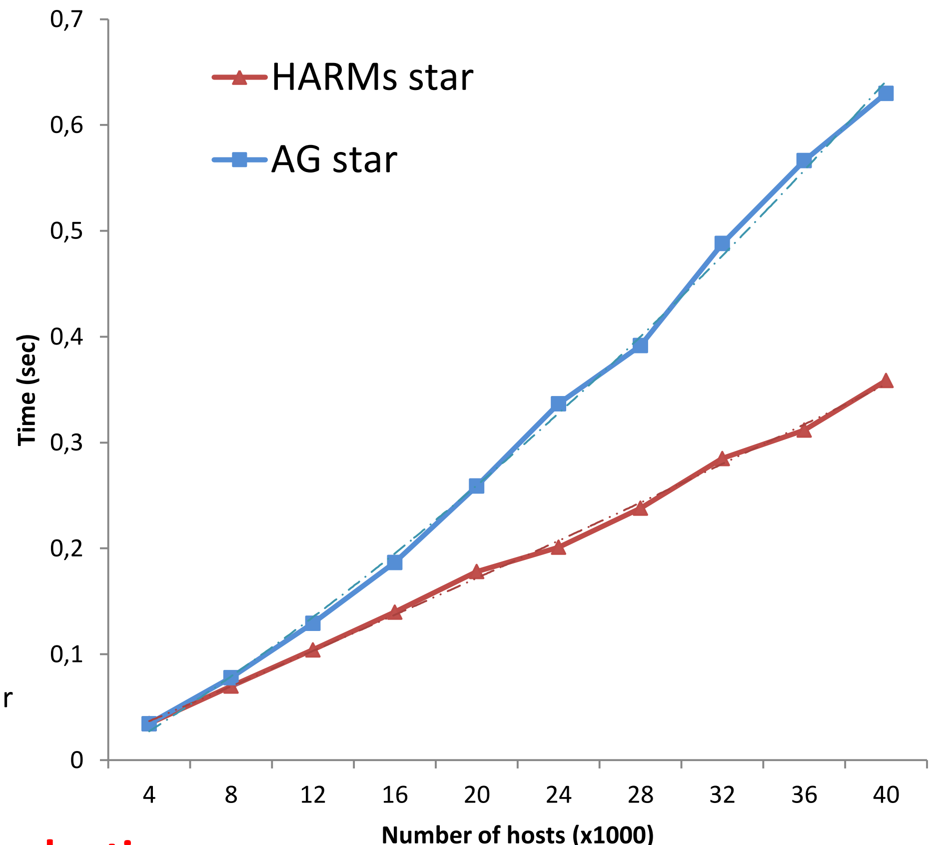
Performance Evaluation via Simulations

Increase #hosts.

HARM performs better than AG in all topology types.

- Simulation 2 – various topologies, attack path length unbounded





Evaluation

Fixed No. of Vulnerabilities = 10

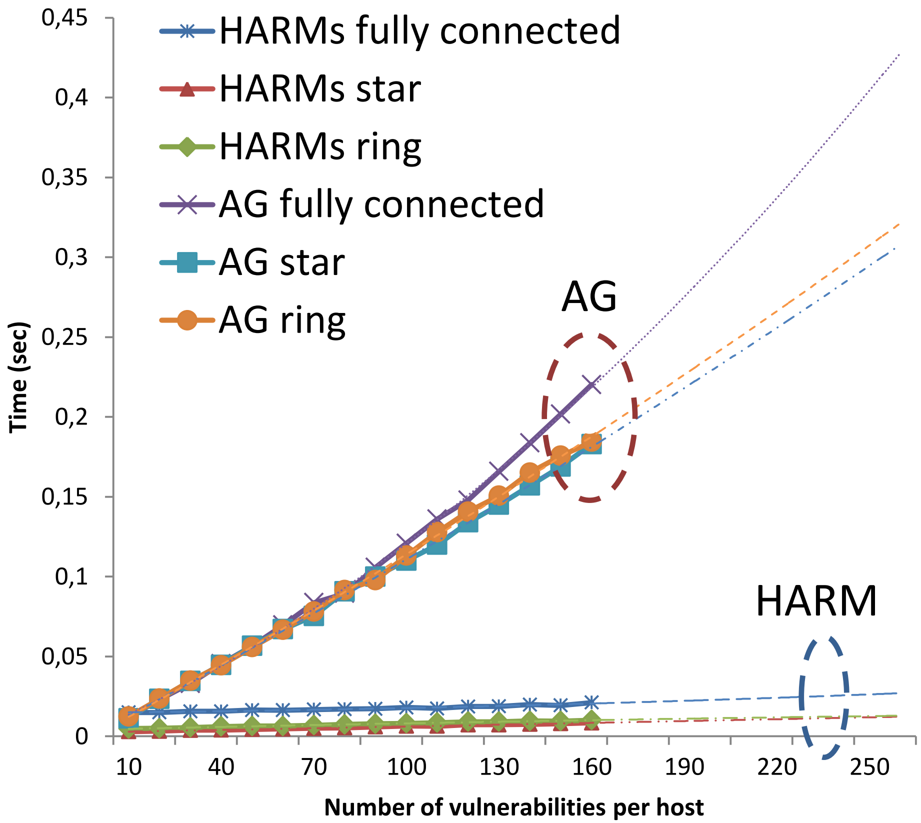
(1 remote-to-root, 9 remote-to-other)

Performance Evaluation via Simulations

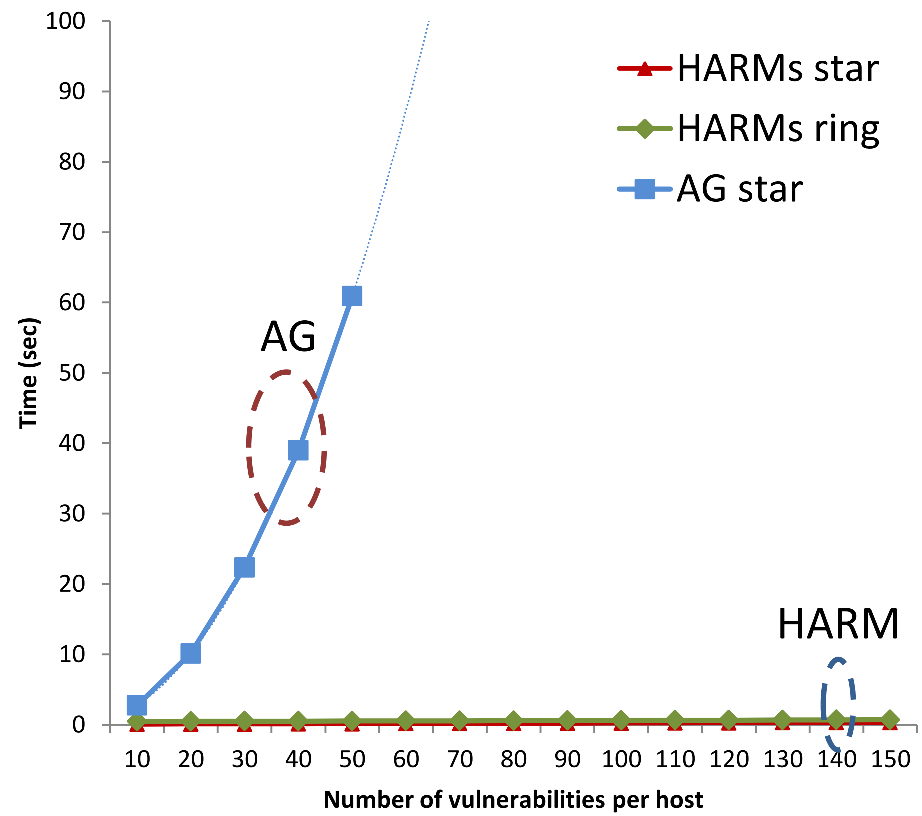
Increase #vul.

HARM performs better than AG in all topology types.

- Simulation 3 – various number of vulnerabilities (L2R only), attack path length unbounded



Evaluation (Fixed No. of **Hosts = 3**)



Evaluation (Fixed No. of Hosts = 1200)

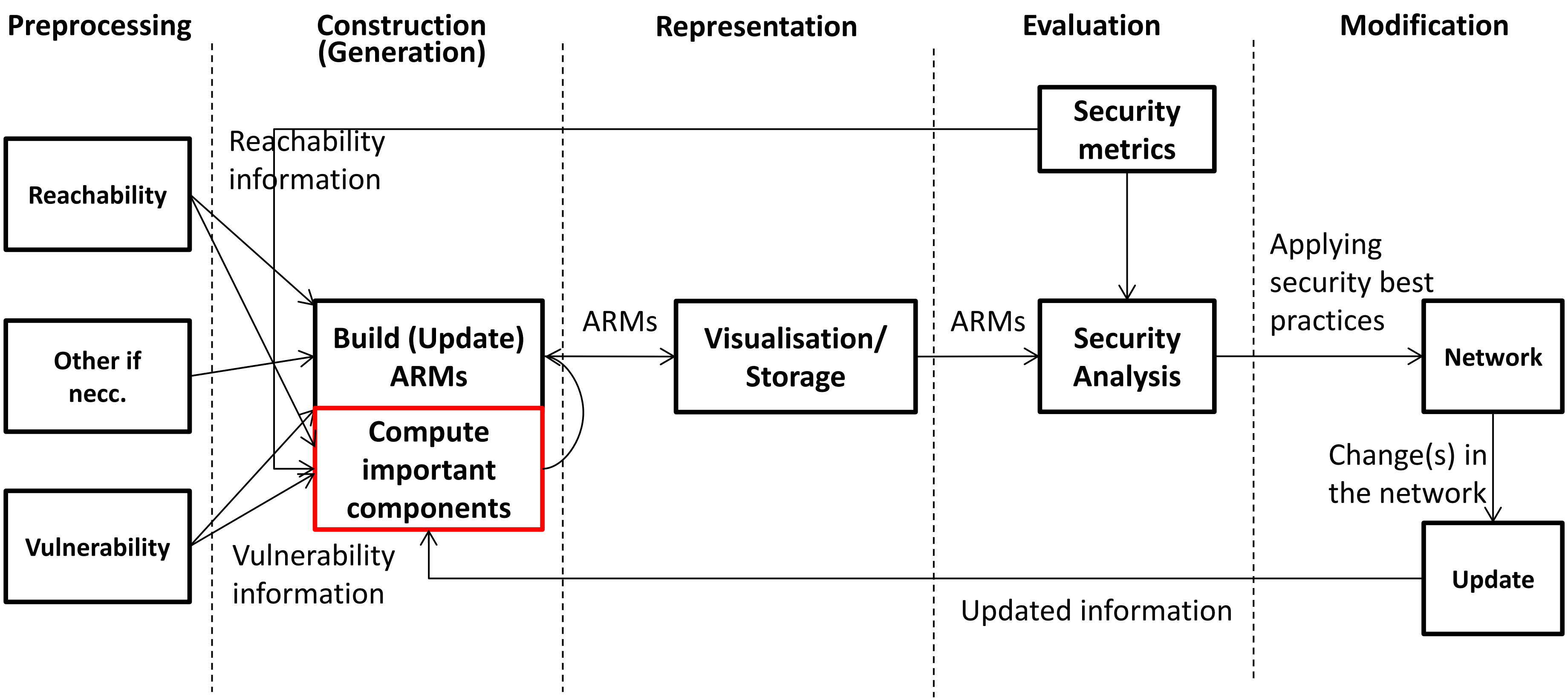
Dealing with Scalability

1. Using Hierarchical ARMs (HARMs)
   - Modelling hosts and vulnerabilities in two different layers (i.e., 2-level hierarchy).
   - Simulation result
2. Construct ARMs based on Important components
   - Improve the construction complexity using less components.
3. Security Analysis based on Important components
   - Using important hosts and vulnerabilities for security analysis.

Construct ARMs using Important Components

- When analysing network security, there are only a **subset** of network components that have a critical role in an event of an attack.
- **All** network components are considered when the ARMs are constructed.
- To **improve** the performance of both construction and evaluation phases of ARMs, we consider to use only important hosts and vulnerabilities.

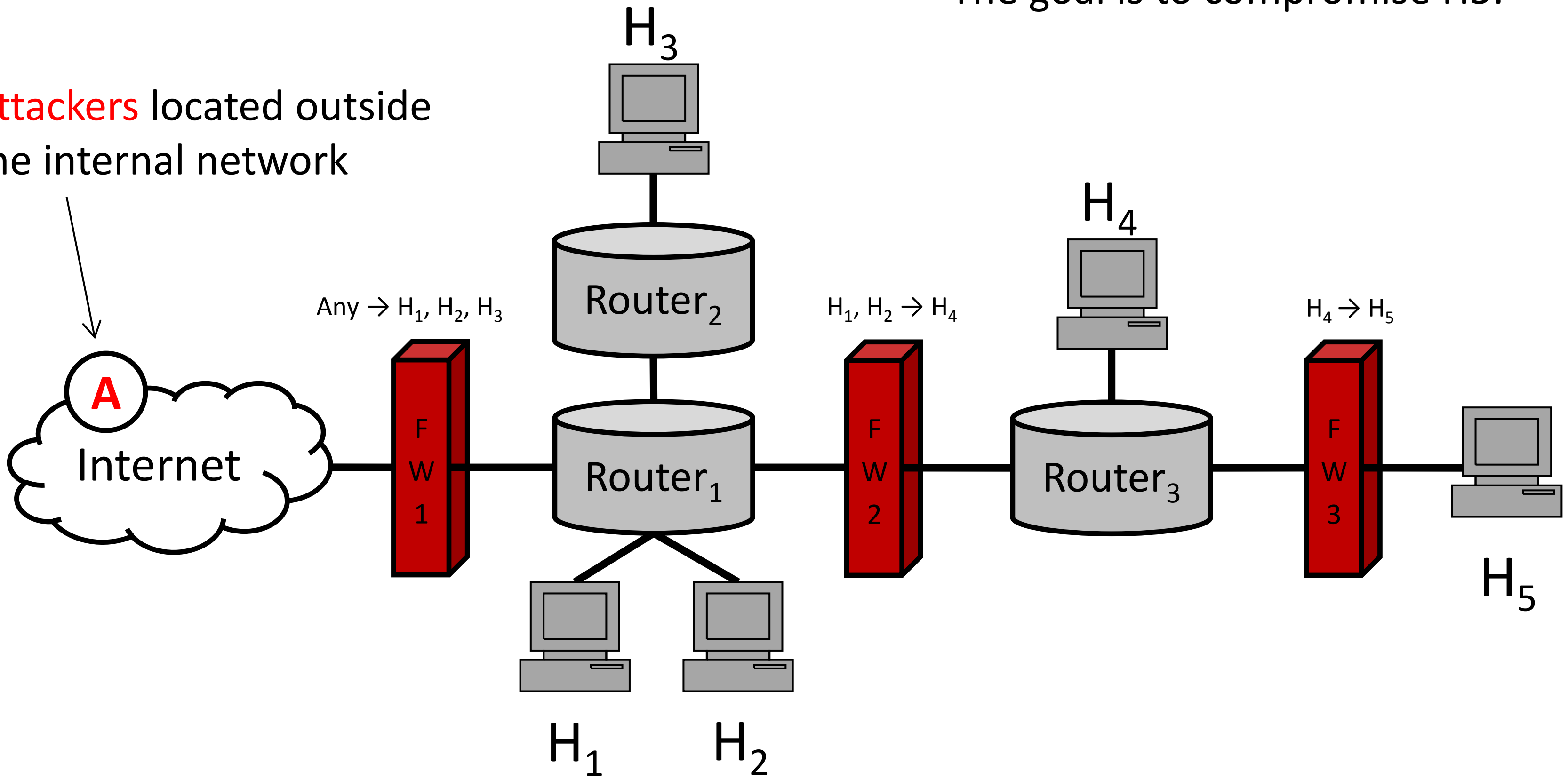
Recap – ARM life cycles



An example network

The goal is to compromise H5.

Attackers located outside the internal network



An example network and its vulnerabilities

Vulnerabilities of H_1 – H_4

ID	CVE ID	CVSS BS	Impact	Exploitability	CI	Access Level	Authentication
v_1	CVE-2005-1794	6.4	4.9	10.0	P	None	None
v_2	CVE-2011-0661	10.0	10.0	10.0	C	None	None
v_3	CVE-2010-0231	10.0	10.0	10.0	C	None	None
v_4	CVE-2011-2552	7.8	6.9	10.0	N	None	None
v_5	CVE-1999-0520	6.4	4.9	10.0	P	None	None
v_6	CVE-2010-2729	9.3	10.0	8.6	C	None	None
v_7	CVE-1999-0505	7.2	10.0	3.9	C	Admin	None
v_8	CVE-2002-1117	5.0	2.9	10.0	P	None	None
v_9	CVE-2003-0386	4.3	2.9	8.6	P	None	None
v_{10}	CVE-2010-0025	5.0	2.9	10.0	P	None	None
v_{11}	CVE-1999-0497	0.0	0.0	10.0	N	None	None

Vulnerabilities of H_5

ID	CVE ID	CVSS BS	Impact	Exploitability	CI	Access Level	Authentication
v_{12}	CVE-2011-1789	5.0	2.9	10.0	N	None	None
v_{13}	CVE-2011-1786	5.0	2.9	10.0	N	None	None
v_{14}	CVE-2011-1785	7.8	6.9	10.0	N	None	None
v_{15}	CVE-2011-0355	7.8	6.9	10.0	N	None	None
v_{16}	CVE-2010-4573	9.3	10.0	8.6	C	None	None
v_{17}	CVE-2010-3609	5.0	2.9	10.0	N	None	None
v_{18}	CVE-2010-1142	8.5	10.0	6.8	C	None	Single System
v_{19}	CVE-2010-1141	8.5	10.0	6.8	C	None	Single System
v_{20}	CVE-2009-3733	5.0	2.9	10.0	P	None	None
v_{21}	CVE-2008-4281	9.3	10.0	8.6	C	None	None
v_{22}	CVE-2008-2097	9.0	10.0	8.0	C	Admin	Single System

using vulnerability scanners such as NESSUS

CI: confidentiality impact

Ranking hosts

- Ranking Hosts w.r.t NCMs

	Degree	Closeness	Betweenness	Rank Sum	Final Rank
H_4	3/4	4/5	10/12	3	1
H_1	3/4	4/5	8/12	4	2
H_2	3/4	4/5	8/12	4	2
H_3	2/4	4/7	4/12	12	4
H_5	1/4	4/12	4/12	14	5

  – Degree (node popularity), Closeness (related distance), Betweenness (usage of a node between paths)
- combine all NCMs to formulate the final rank
  – Each rank acted as a score to give the final rank (i.e., scores are used to re-rank nodes)
  – Rankings from each NCM are used as the importance score

Ranking vulnerabilities on hosts

- Ranking Vulnerabilities

Vulnerabilities of H_1 – H_4

	v_2	v_3	v_6	v_4	v_7	v_1	v_5	v_8	v_{10}	v_9	v_{11}
CVSS BS	10.0	10.0	9.3	7.8	7.2	6.4	6.4	5.0	5.0	4.3	0.0
Rank	1	1	3	4	5	6	6	8	8	10	11

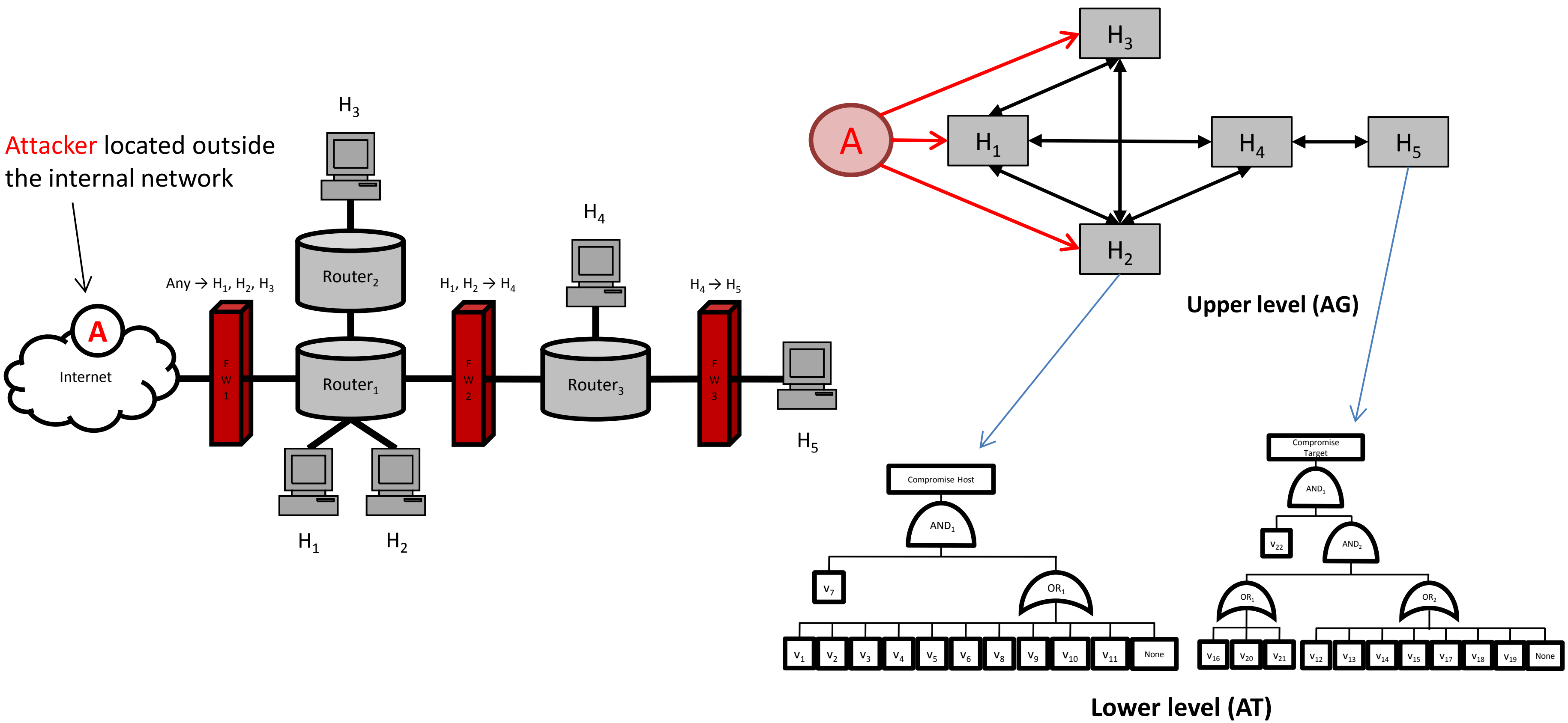
Vulnerabilities of H_5

	v_{16}	v_{21}	v_{22}	v_{18}	v_{19}	v_{14}	v_{15}	v_{12}	V_{13}	v_{17}	v_{20}
CVSS BS	9.3	9.3	9.0	8.5	8.5	7.8	7.8	5.0	5.0	5.0	5.0
Rank	1	1	3	4	4	6	6	8	8	8	8

Vulnerabilities are ranked based on their CVSS BSs (common vulnerability score system base score)

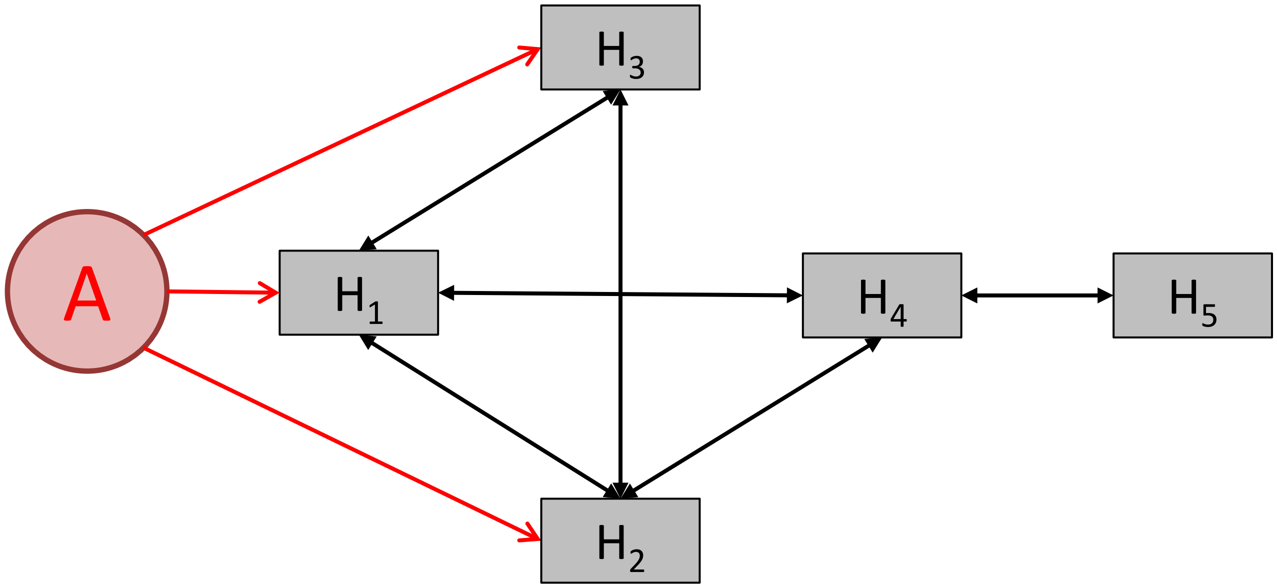
Important vulnerabilities are selected based on the threshold value (e.g., higher than the average CVSS BSs)

Revisit: the example network



A HARM for the example net

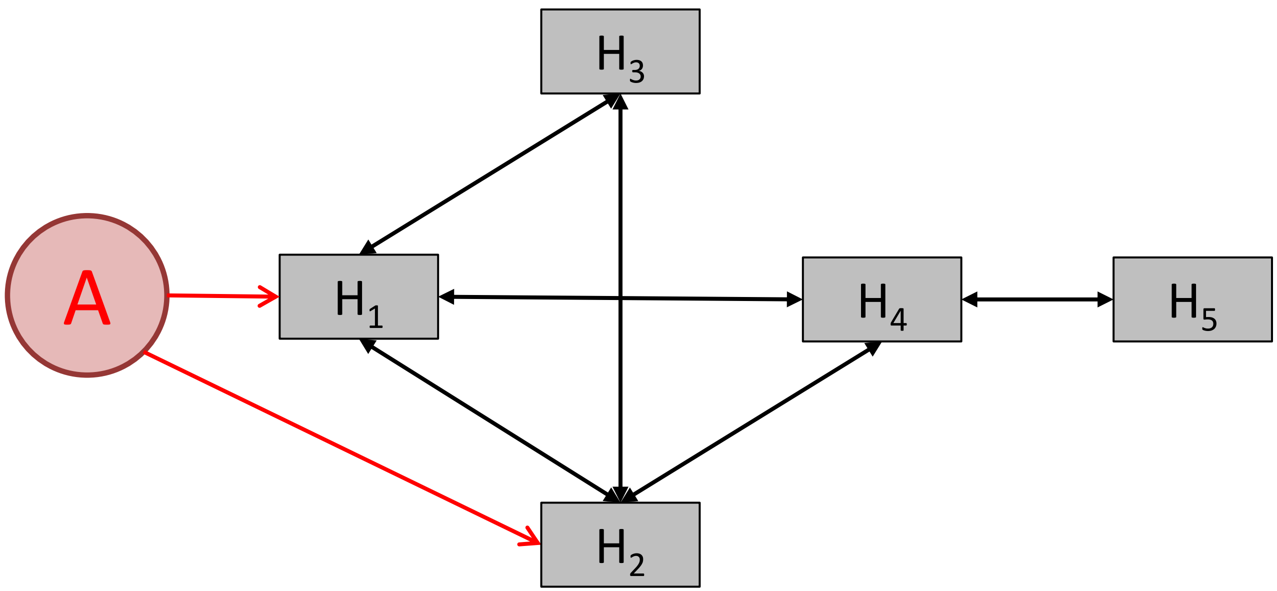
- Naïve method : AG-AT HARM – upper level



Upper level (AG)

A simplified HARM

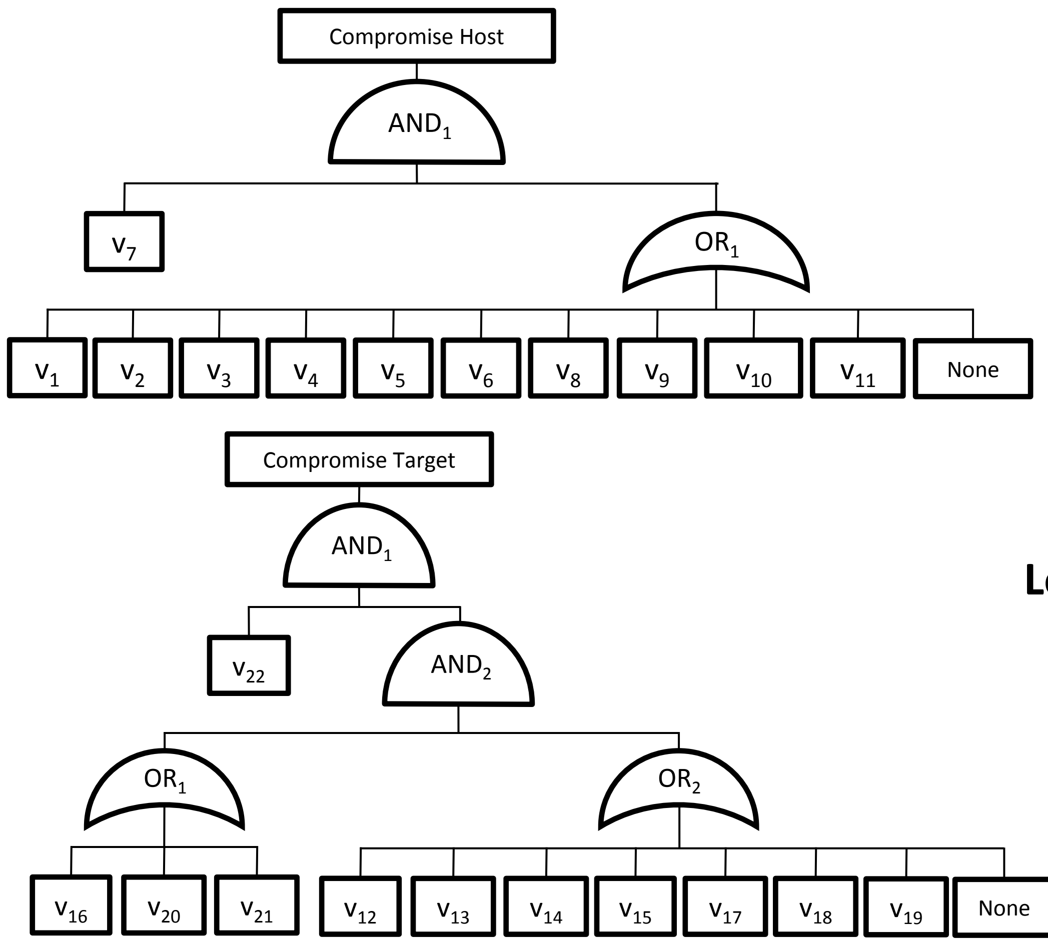
- Using only important hosts : AG-AT HARM in the upper level



Upper level (AG)
Using 3 hosts

A HARM for the example net

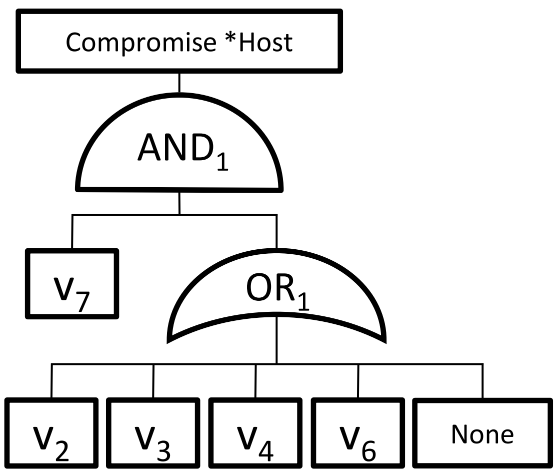
- Naïve method : AG-AT HARM in the lower level



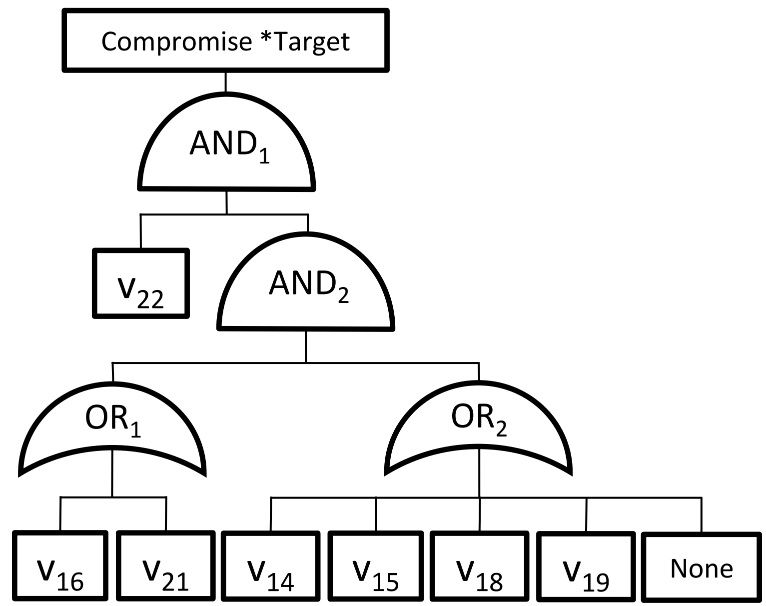
Lower level (AT)

A simplified HARM

- Using Important vulnerabilities: AG-AT HARM



Using 5 vulnerabilities

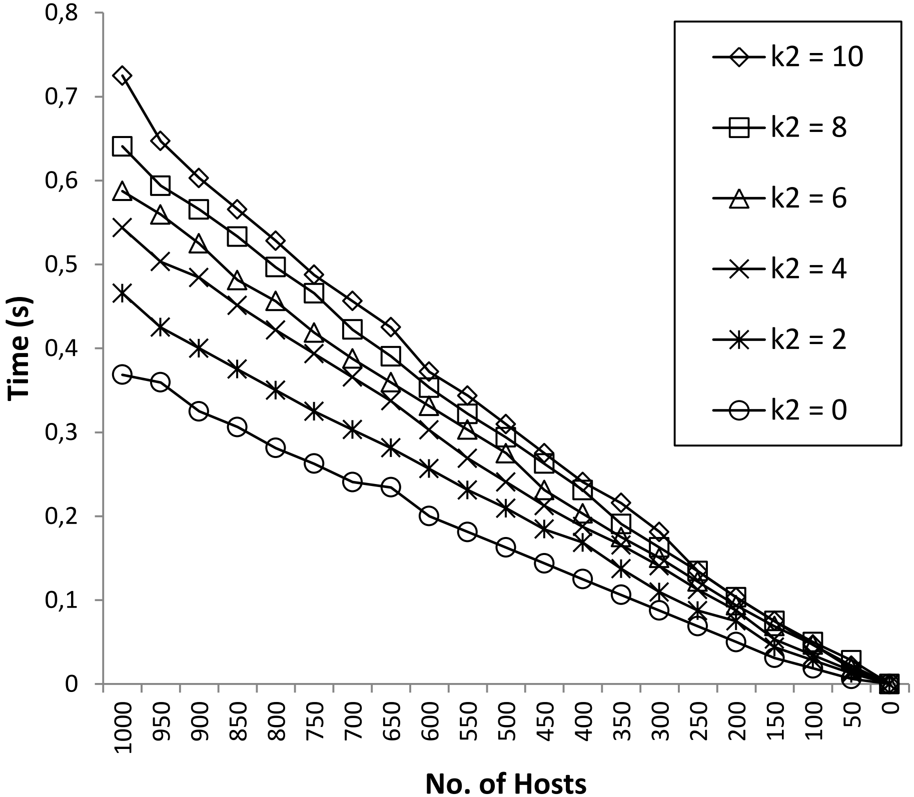


Using 7 vulnerabilities

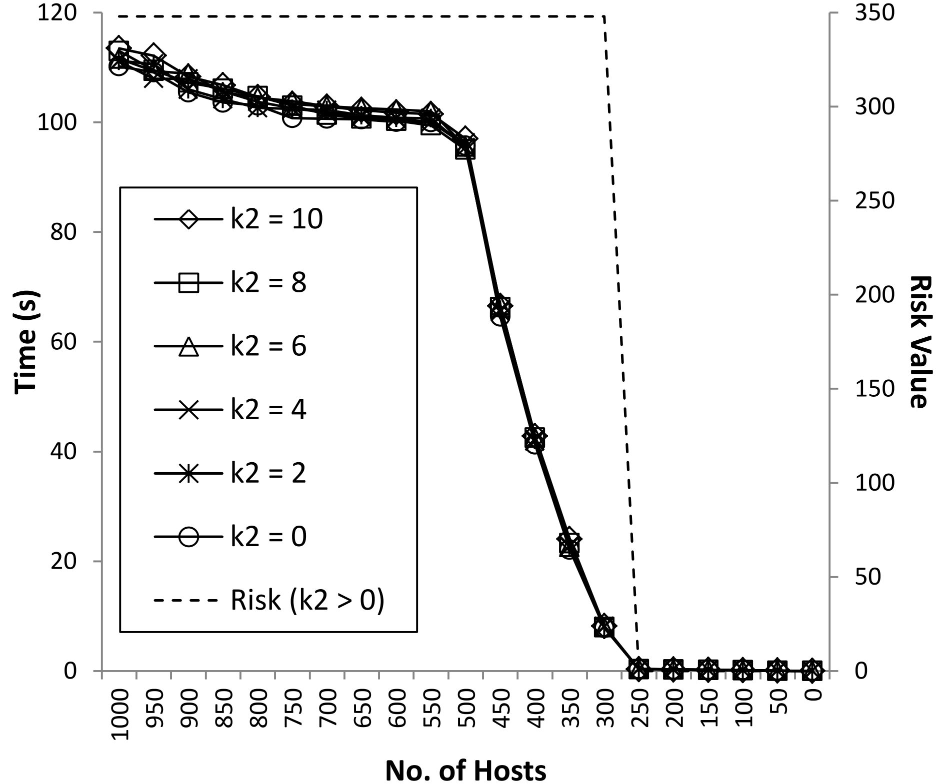
Lower level (AT)

Performance Simulation

- Results – host based security analysis



Construction
(with *k2* number of vulnerabilities)

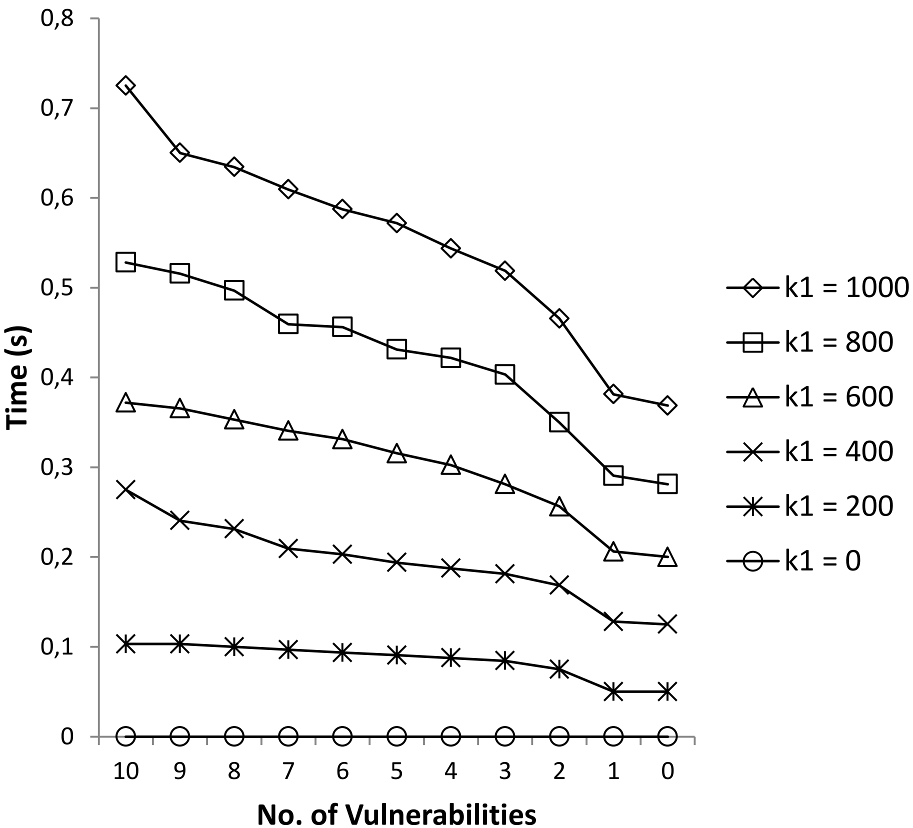


Evaluation
(with *k2* number of vulnerabilities)

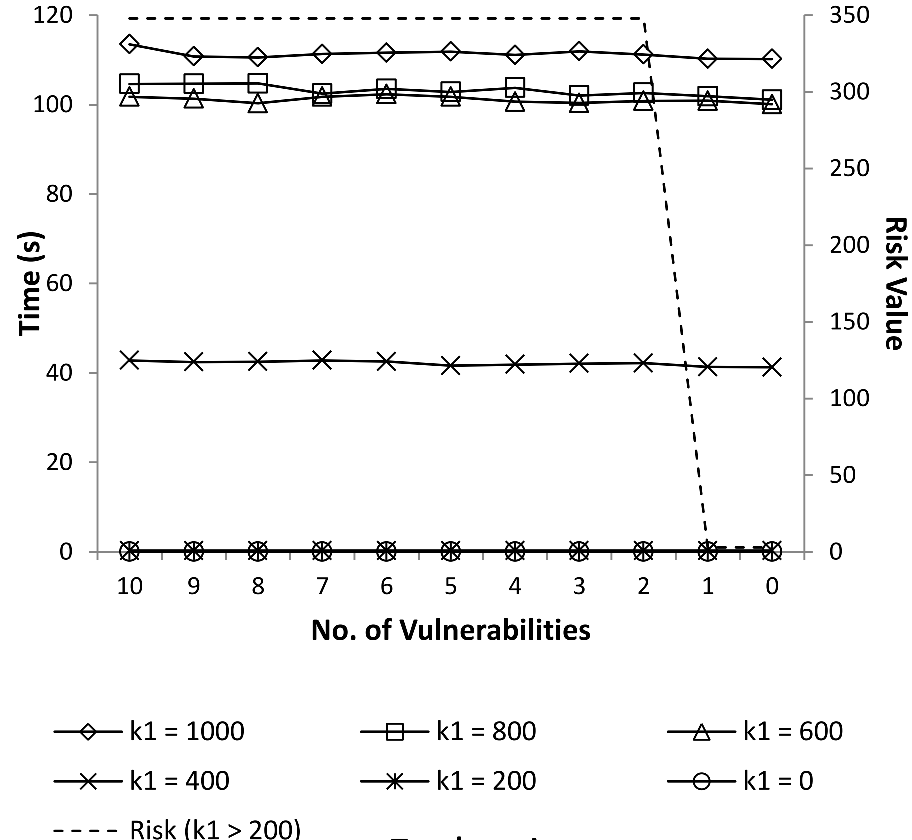
The construction time linearly improves as the number of important hosts modelled reduce.
For evaluation, there is a steady improvement until the host number reaches 500.
From then, it rapidly improves the performance.

Performance Simulation (cont.)

- Results – vulnerability based security analysis



Construction
(with *k1* number of hosts)



Evaluation
(with *k1* number of hosts)

The construction time linearly improves as the number of important vuls modelled reduce.
The variation of vulnerability numbers has minimum effect

Conclusion

- Constructing ARMs using only **important** hosts and vulnerability can **improve** the performance in construction and evaluation.
  - Nearly equivalent security analysis is performed, with 87% **improved** construction time and 99.5% **improved** evaluation time in the simulation.

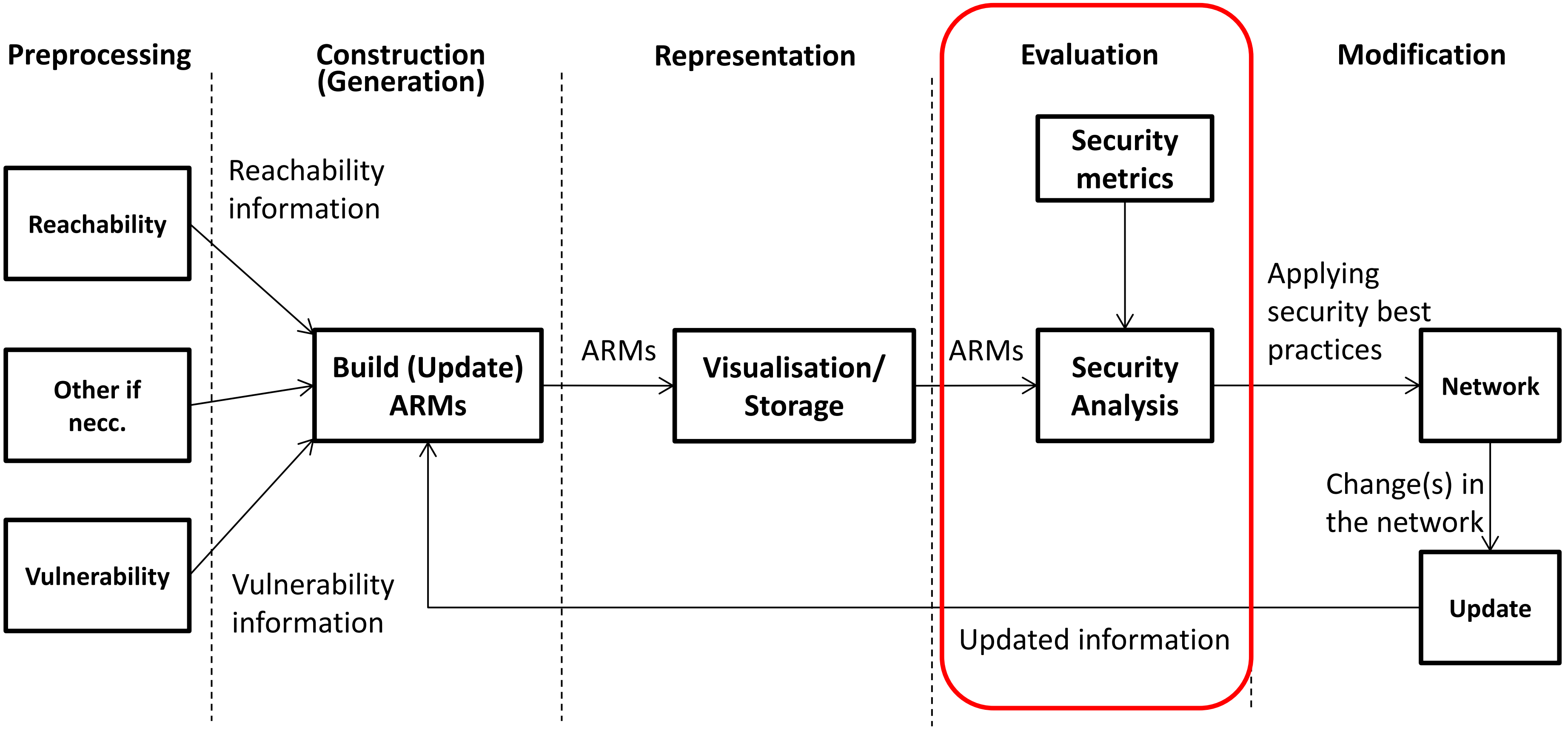
Dealing with Scalability

1. Using Hierarchical ARMs (HARMs)
   - Modelling hosts and vulnerabilities in two different layers (i.e., 2-level hierarchy).
   - (semi-)automated generation
   - Simulation result

2. Construct ARMs based on Important components
   - Improve the construction complexity using less components.

3. Security Analysis based on Important components
   - Using important hosts and vulnerabilities for security analysis.

Attack Representation Model (ARM) life cycles

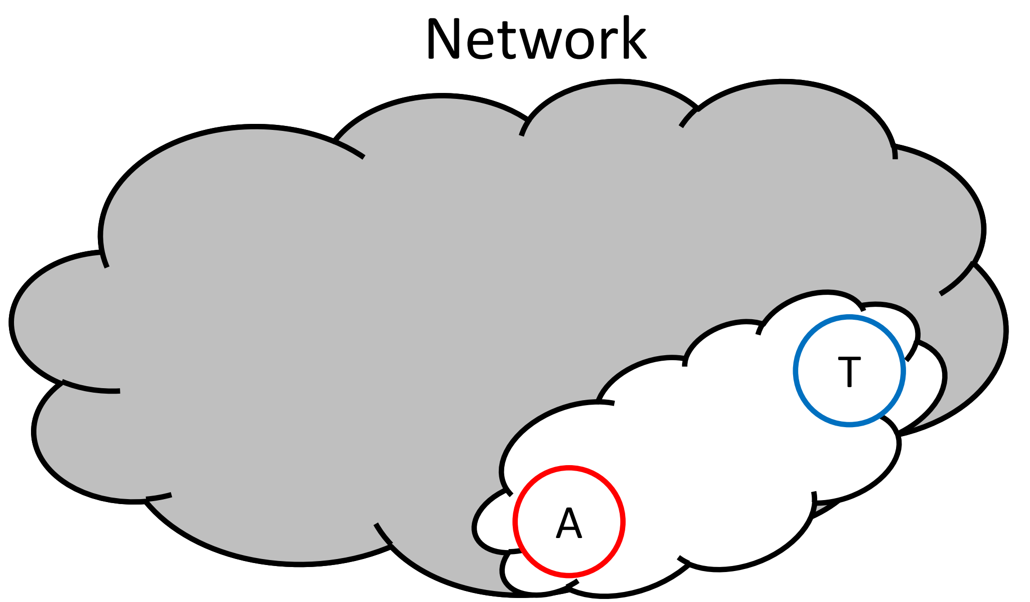


1. Use only important hosts
2. Use only important vulnerabilities in hosts

1. Scalable?
2. Equivalent security solution c.f. exhaustive search?

Network coverage

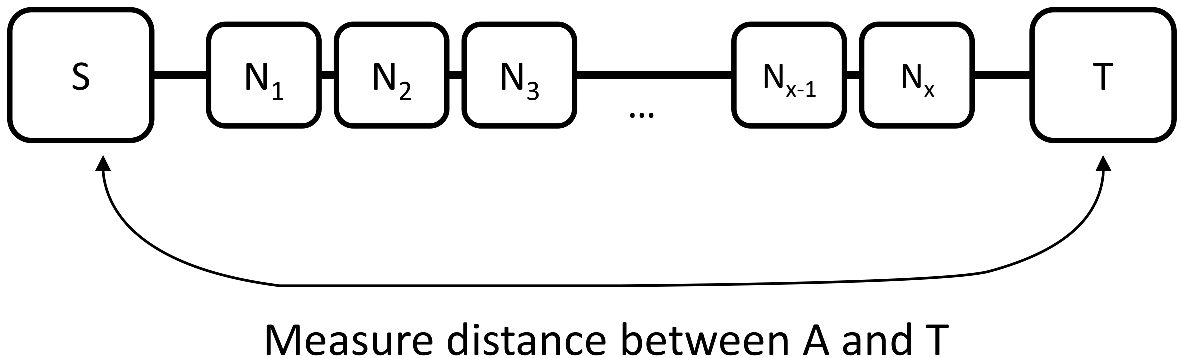
- Consider an attack scenario that covers only a subset of the network (e.g., an attacker located inside the network).

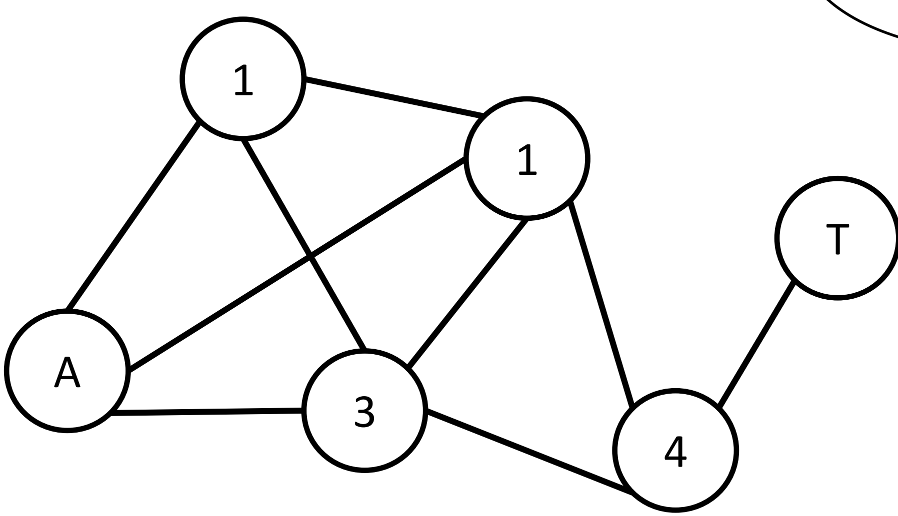


How to define the subnet covered by the attack scenario?

Attacker to Victim Centrality (AVC)

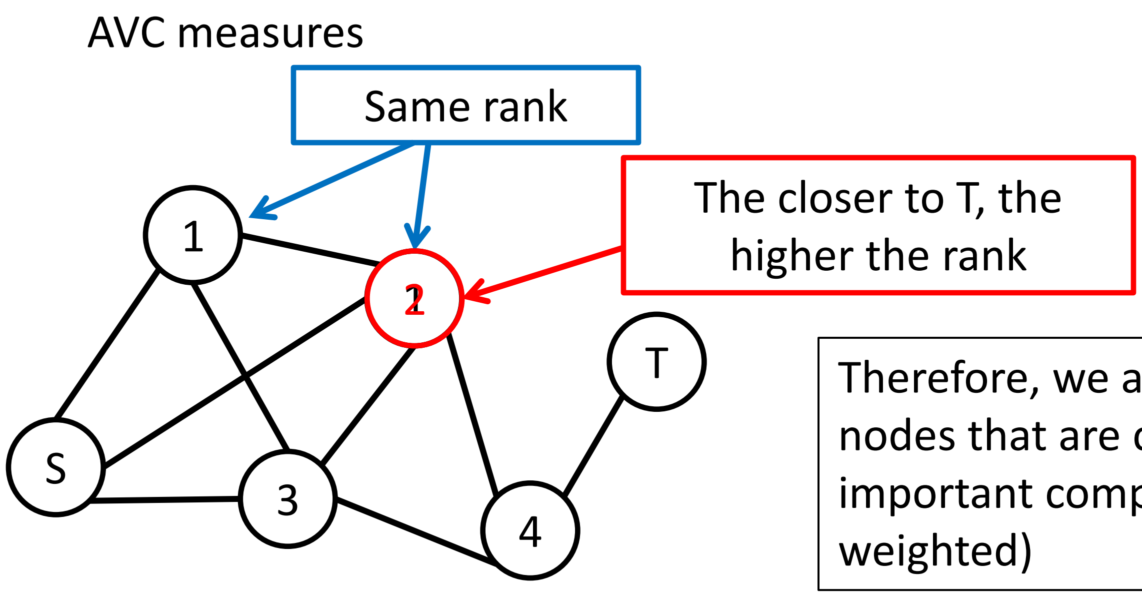
- Typical NCMs in the upper level do not consider the location of the attacker and the target (victim).
- We define a location-based (Attacker to Victim) centrality (AVC) measurement based on distance measurements.





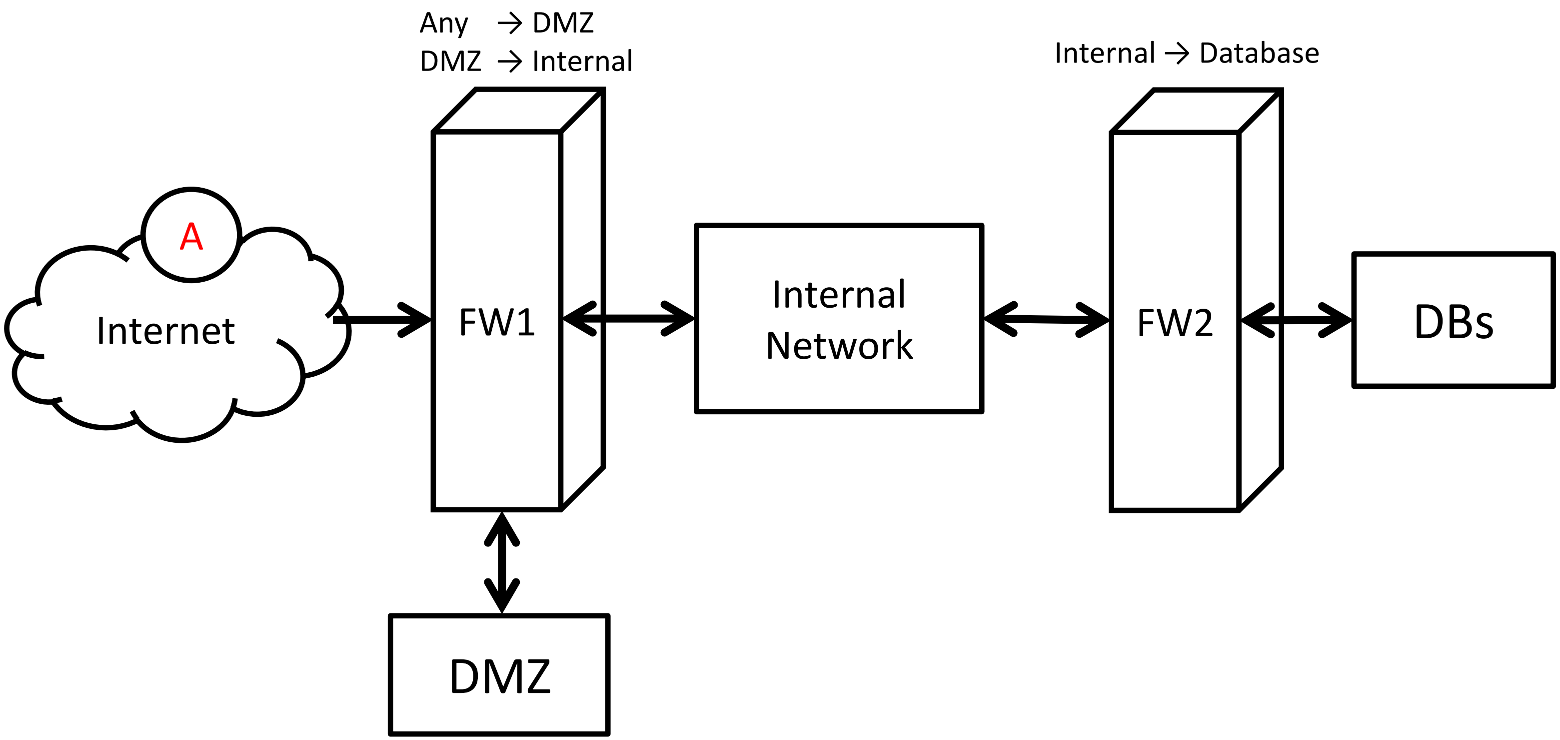
Attacker to Victim Neighbour Centrality (AVNC)

- If there are components with the same ranking, then the AVC may not identify important components correctly.



Therefore, we assign more weights to nodes that are closer to highly important components (value could be weighted)

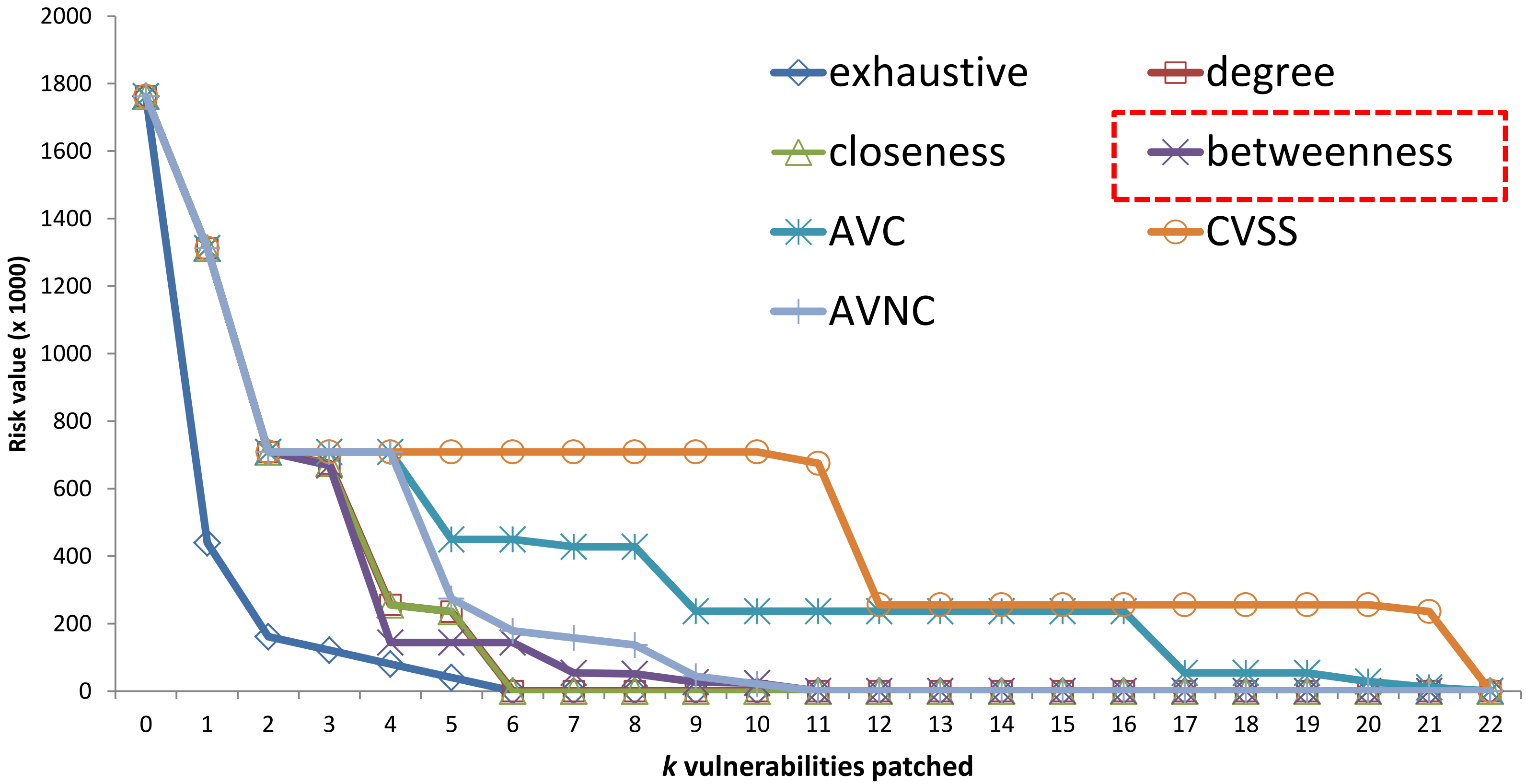
Security Analysis via Simulation



Attack scenario:
Attacker compromise x_i DMZ hosts, through Internal hosts, then finally obtain data from the designated DB

Security Analysis via Simulation (cont.)

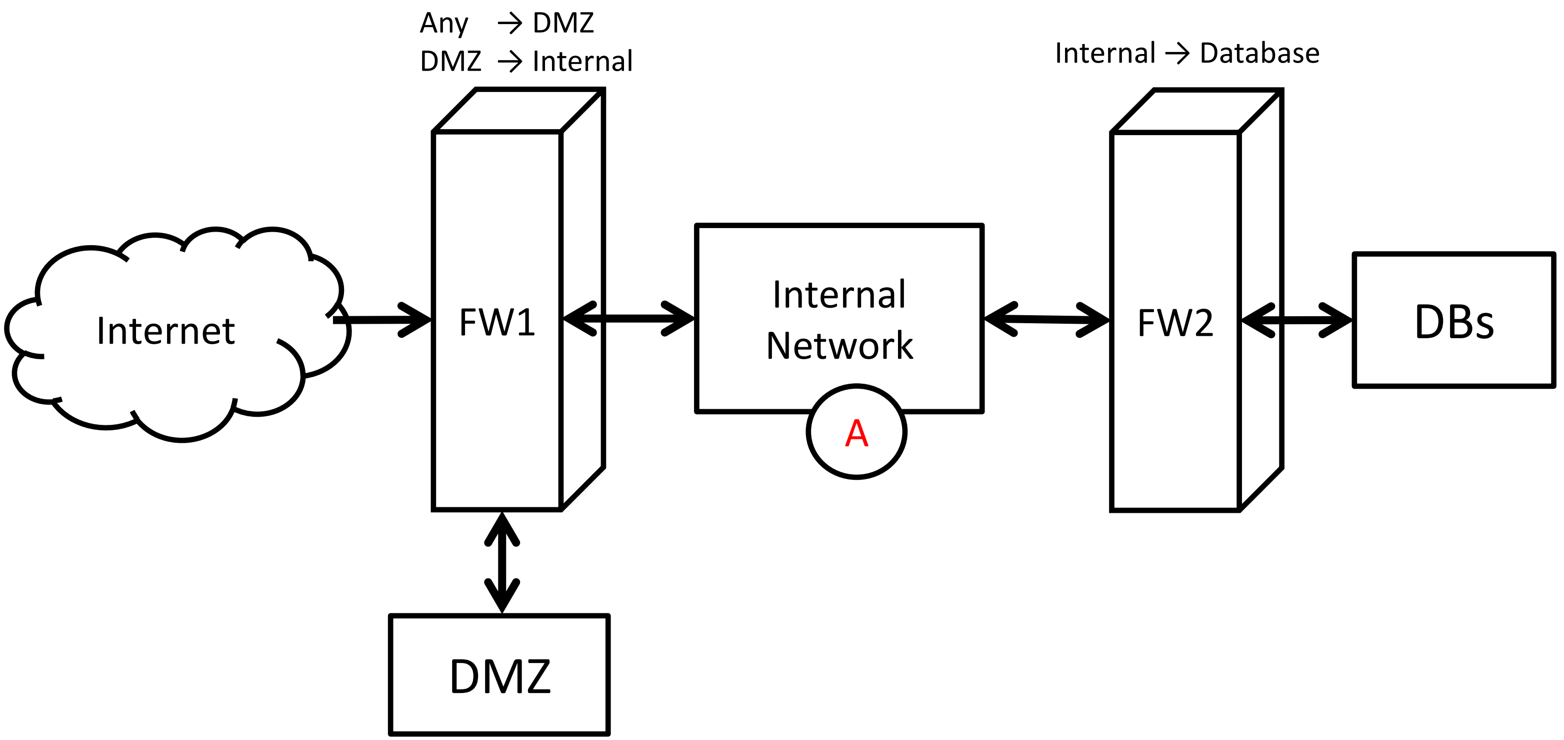
- Results – multiple subnets, external attacker



The host based importance based measures perform better than vulnerability based importance measures

There are components with same importance rankings. Patching vulnerabilities for these components may not reduce the risk value, so there are fluctuations in the graph.

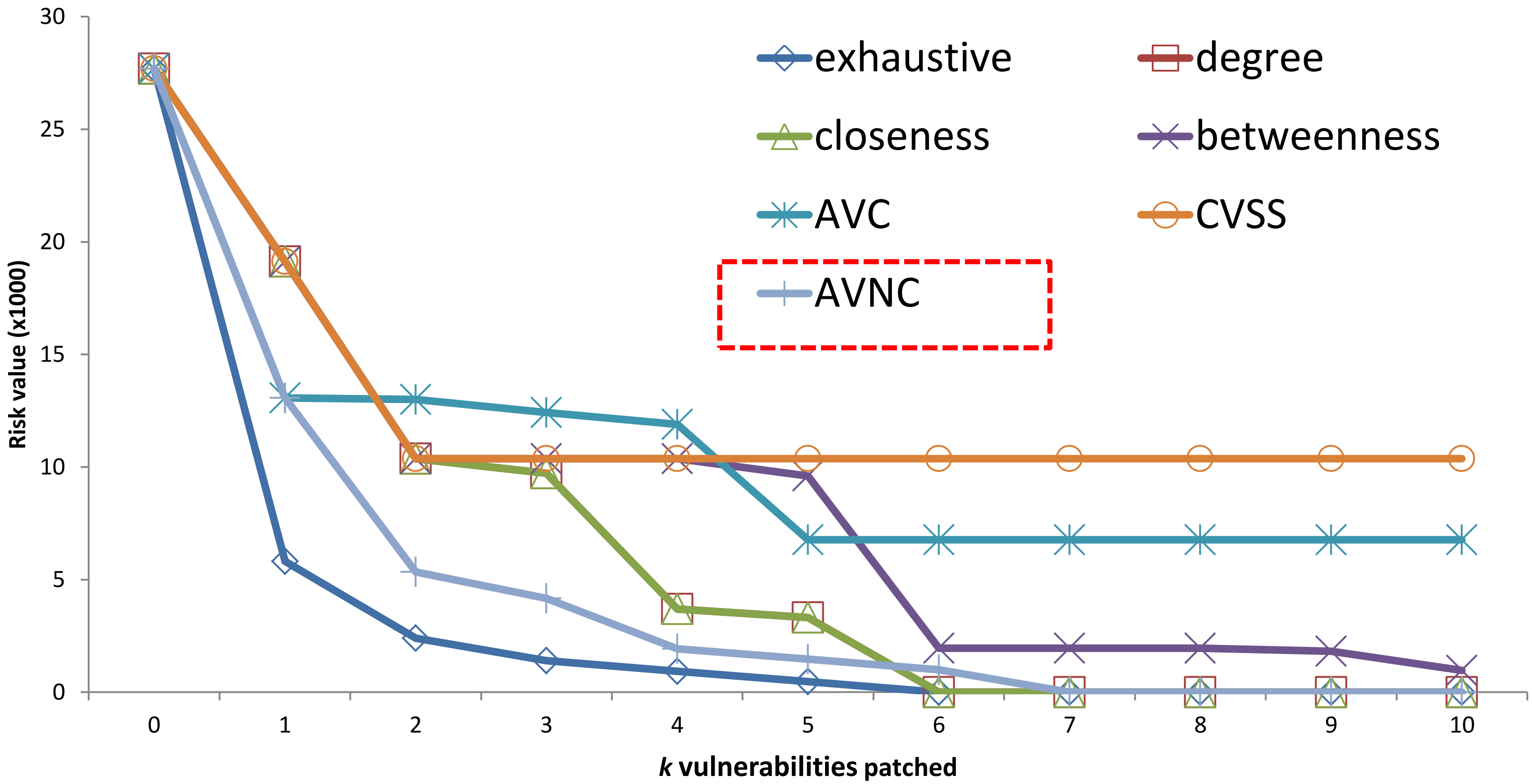
Security Analysis via Simulation (cont.)



Attack scenario:
Attacker compromise x_i Internal hosts, obtain data from the designated DB

Security Analysis via Simulation (cont.)

- Results – multiple subnets, internal attacker



The location based centrality measure AVNC performs most closely to the exhaustive search.

Limitations

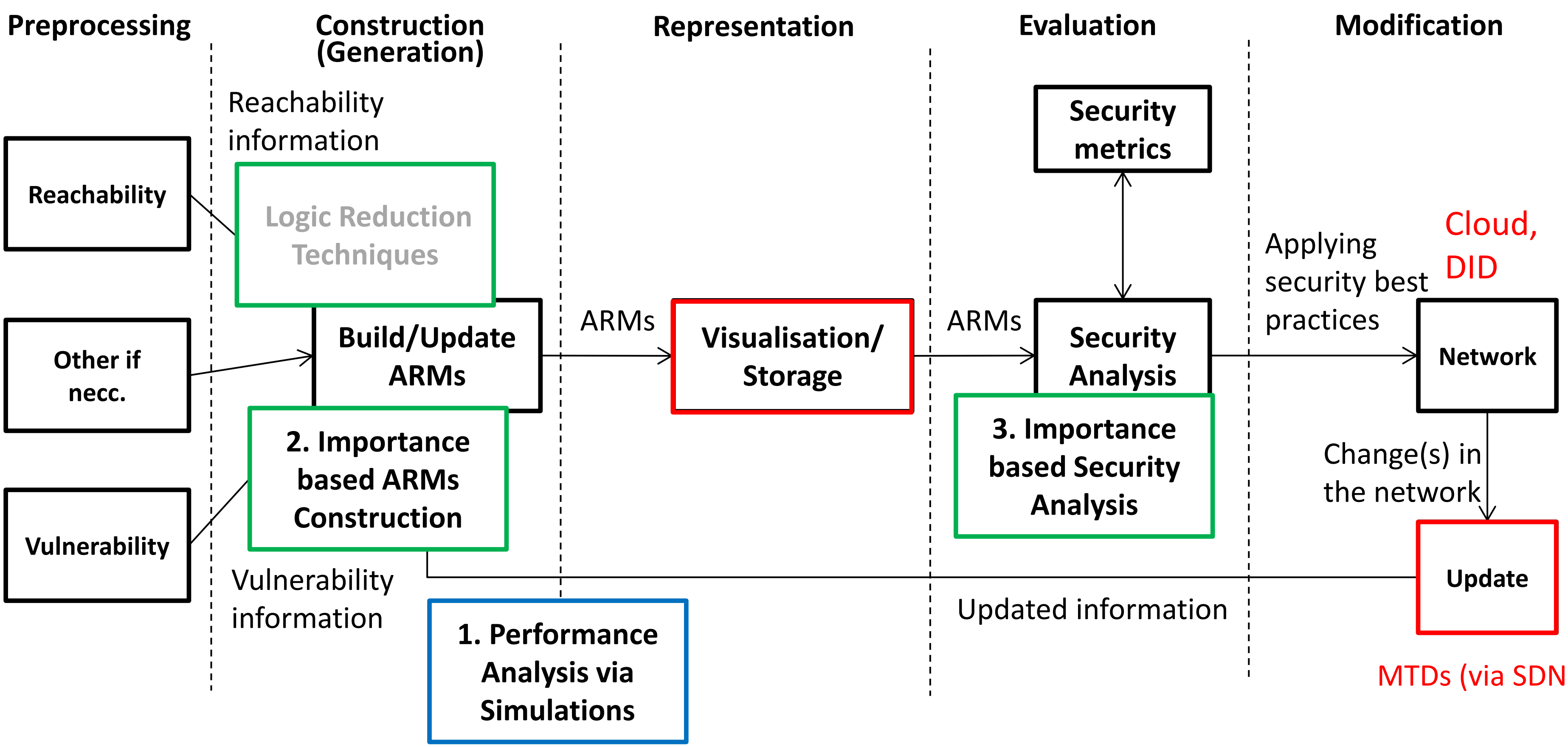
- **Combinations** of rankings
  - Overlaps between NCMs indicate improvements can be achieved by combining their rankings
  - Combining with vulnerability rankings
- **Multiple** target host locations
  - Changes in rankings
- Attack on **less** important hosts and vulnerabilities
  - High cost attacks
  - Advanced persistent threat (APT)

Conclusion

- Evaluating HARMs using only **important** hosts and vulnerability can **improve** the performance of evaluation.

- Nearly equivalent security analysis is achievable, with **improved** evaluation time (from exponential down to polynomial.)

Final summary



Scalable Security Models

Hagley Park, Christchurch, New Zealand

Dong-Seong Kim
dongseong.kim@canterbury.ac.nz
University of Canterbury

Related publications

- Arpan Roy, Dong Seong Kim, Kishor S. Trivedi: Attack countermeasure trees (ACT): towards unifying the constructs of attack and defense trees. Security and Communication Networks 5(8): 929-943 (2012)
- Arpan Roy, Dong Seong Kim, Kishor S. Trivedi: Scalable optimal countermeasure selection using implicit enumeration on attack countermeasure trees. DSN 2012
- Jin Hong, Dong Seong Kim,"HARMs: Hierarchical Attack Representation Models for Network Security Analysis" in Proc. of the 10th Australian Information Security Management Conference (SECAU 2012)
- Jin Hong, Dong Seong Kim, "Performance analysis of scalable attack representation models" In Proc. of the 28th IFIP TC-11 International Information Security and Privacy Conference (SEC 2013)
- Jin Hong, Dong Seong Kim, Scalable Security Analysis in Hierarchical Attack Representation Model using Centrality Measures, in Proc. of RSDA 2013 in conjunction with DSN 2013.
- Jin Hong, Dong Seong Kim, Scalable Attack Representation Model using Logic Reduction Techniques, in Proc. of TrustCom 2013.
- Jin Hong, Dong Seong Kim, Scalable Security Model Generation and Analysis using k-importance Measures, in Proc. of SecureComm 2013.