INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Agent Applications in Network Security Monitoring

Martin Rehak, Department of Cybernetics, CTU in Prague

Tato prezentace je spolufinancována Evropským sociálním fondem a státním rozpočtem České republiky.

Part 1: (In)secure Monitoring

Security monitoring systems tend to get more and more autonomy due to the:

- Cost restrictions
- Exponential growth of coverage
- Data fusion options
- Decision quality

Their pervasiveness makes them a high-value target for the attacker

Security Monitoring Goals

Defense in depth – one layer of defense is not sufficient for high-value targets

Progressive layers of defense need to be configured to reflect the (a) **business needs** and (b) **threat environment**

Exploitation time – monitoring should ensure that the attacker would not be able to profit from any successful exploit of individual component

Example

Door inspection: High-security locks (used to) resist a professional attacker for at least 10 min.

Monitoring system needs to ensure that nobody can attack the doors for 10 minutes without being **observed**, **detected** and **identified**.

… and this is subject to multitude of attacks on monitoring, that can be explained using a multi-agent paradigm

Attacks on Monitoring

Approach/Target	Sensor	Feature extraction/ Pattern matching	Detection model
Passive	Passive sensor attacks	Passive attacks on pattern matching	Passive attacks on detection model
Active	Active sensor attacks	Active attacks on pattern matching	Active attacks on detection model
Denial of service	Floods the operator with false alarms		

Classification follows (simplifies and re-formulates) [*Barreno 2006*]

Additional axis: targeted and indiscriminate

Passive Attacks

Probe capabilities, resolution and current state of various system components in order to determine their capabilities and avoid them in the future

Target:

- Sensor (*host IDS fingerprinting, side channel attacks on IDS*)
- Feature extraction (*polymorphic malware, shape-disrupting camouflage*)
- Learning and Model (*learning, adaptive malware*)

Active Attacks

Shape algorithm inputs in order to **influence** its capability to detect intruder’s future actions

Targets:

- Sensor (jamming, electronic warfare attacks)
- Feature extraction (*concurrent attacks in network security*)
- Model and learning (*adversarial network traffic shaping*)

Increasing Sophistication

Improving computational capacities allow the attackers to target the internal state of the model, rather than only sensor or pattern matching as before

Automated attack methods can be based on machine learning, genetic algorithms or a combination of thereof

Example: *Self-learning malware for automatic avoidance of IDS system*

CAMNEP Goals & Assumptions

Improve the **error rate**

- lower false positives
- same false negatives

Validation & Stability

- reliable response to typical threats
- traffic-independent response

Management

- self-optimization and self-configuration

Reasonable-sized traffic

Reasonable attack types

Not-real time (minimal response delay 40 sec.)

Integrates with other defense techniques

Low predictability by opponent

Structured, actionable output

CAMNEP: System overview

Date flow start	Duration	Proto	Src IP Addr:Port		Dst IP Addr:Port	Flags	Tos	Packets	Bytes	pps	bps	Bpp	Flows
2007-02-12 10:00:43.839	0.001	TCP	147.251.192.4:524	->	147.251.192.86:1033	.AP...	0	2	187	1999	1.4 M	93	1
2007-02-12 10:00:43.845	0.000	UDP	147.251.192.1:53	->	147.251.192.153:1159	......	0	1	133	0	0	133	1
2007-02-12 10:00:43.848	0.000	UDP	147.251.192.153:1159	->	147.251.192.1:53	......	0	1	53	0	0	53	1
2007-02-12 10:00:43.848	0.000	TCP	147.251.193.76:1028	->	147.251.192.4:524	.AP...	0	2	284	0	0	142	1
2007-02-12 10:00:43.848	0.000	UDP	147.251.193.76:1529	->	147.251.192.1:53	......	0	2	126	0	0	63	1
2007-02-12 10:00:43.848	0.000	UDP	147.251.192.1:53	->	147.251.193.76:1529	......	0	2	224	0	0	112	1
2007-02-12 10:00:41.525	2.324	TCP	147.251.193.76:1026	->	147.251.192.4:524	.AP...	0	3	292	1	1005	97	1
2007-02-12 10:00:43.848	0.001	TCP	147.251.192.4:524	->	147.251.193.76:1028	.AP...	0	2	162	1999	1.2 M	81	1
2007-02-12 10:00:43.849	0.000	UDP	147.251.4.33:53	->	147.251.193.76:1529	......	0	2	224	0	0	112	1
2007-02-12 10:00:41.552	2.298	TCP	147.251.192.4:524	->	147.251.193.76:1026	.AP...	0	2	116	0	403	58	1
2007-02-12 10:00:43.848	0.002	UDP	147.251.193.76:1529	->	147.251.4.33:53	......	0	2	126	999	503999	63	1
2007-02-12 10:00:43.336	0.580	TCP	147.251.192.1:44254	->	70.42.39.14:2703	.AP.SF	0	6	510	10	7034	85	1
2007-02-12 10:00:43.421	0.497	TCP	147.251.192.1:44255	->	70.42.39.14:2703	.AP.SF	0	7	534	14	8595	76	1
2007-02-12 10:00:43.935	0.000	UDP	61.151.252.240:53	->	147.251.192.1:53859	......	0	1	123	0	0	123	1
2007-02-12 10:00:42.632	1.307	TCP	84.60.32.153:24023	->	147.251.192.106:6881	.AP.SF	0	12	2194	9	13429	182	1
2007-02-12 10:00:43.624	0.318	TCP	194.79.52.10:80	->	147.251.192.153:1158	.AP.S.	0	5	3191	15	80276	638	1
2007-02-12 10:00:43.943	0.000	TCP	65.78.80.176:57889	->	147.251.192.106:3514	.A....	0	1	42	0	0	42	1
2007-02-12 10:00:43.943	0.034	TCP	129.12.31.4:14017	->	147.251.195.20:1077	.AP...	0	3	126	88	29647	42	1
2007-02-12 10:00:34.713	9.268	TCP	212.80.76.24:80	->	147.251.192.153:1108	.AP.SF	0	11	6797	1	5867	617	1
2007-02-12 10:00:43.251	0.748	TCP	147.251.192.1:25	->	147.251.4.36:33555	.AP.SF	0	21	1242	28	13283	59	1
2007-02-12 10:00:43.251	0.748	TCP	147.251.4.36:33555	->	147.251.192.1:25	.AP.SF	0	28	28556	37	305411	1019	1
2007-02-12 10:00:44.002	0.004	TCP	194.228.32.6:80	->	147.251.195.144:2887	.A...F	0	2	84	499	167999	42	1
2007-02-12 10:00:43.839	0.197	TCP	147.251.192.86:1033	->	147.251.192.4:524	.AP...	0	4	310	20	12588	77	1
2007-02-12 10:00:44.056	0.000	TCP	194.79.52.199:80	->	147.251.192.153:1160	.AP...	0	2	1150	0	0	575	1
2007-02-12 10:00:43.863	0.197	TCP	147.251.192.153:1160	->	194.79.52.199:80	.AP.S.	0	4	397	20	16121	99	1
2007-02-12 10:00:43.559	0.503	TCP	70.42.39.14:2703	->	147.251.192.1:44255	.APRS.	0	7	399	13	6345	57	1
2007-02-12 10:00:43.467	0.596	TCP	70.42.39.14:2703	->	147.251.192.1:44254	.APRS.	0	8	457	13	6134	57	1
2007-02-12 10:00:23.849	30.141	TCP	147.251.195.20:1077	->	129.12.31.4:14017	.AP...	0	18	780	0	207	43	1
2007-02-12 10:00:44.086	0.000	UDP	147.251.192.1:53859	->	192.42.93.30:53	......	0	1	62	0	0	62	1
2007-02-12 10:00:40.249	3.839	UDP	147.251.192.146:137	->	147.251.193.255:137	......	0	5	370	1	771	74	1
2007-02-12 10:00:43.767	0.000	UDP	82.208.50.129:6354	->	147.251.192.27:32225	......	0	1	119	0	0	119	1
2007-02-12 10:00:43.773	0.000	UDP	82.208.50.129:21128	->	147.251.192.27:29123	......	0	1	119	0	0	119	1
2007-02-12 10:00:43.294	0.486	TCP	147.251.192.153:1158	->	194.79.52.10:80	.AP.S.	0	5	459	10	7555	91	1
2007-02-12 10:00:43.151	0.656	TCP	147.251.192.5:80	->	147.251.193.59:1414	.AP.SF	0	58	71055	88	866524	1225	1
2007-02-12 10:00:43.149	0.658	TCP	147.251.193.59:1414	->	147.251.192.5:80	.AP.SF	0	32	2144	48	26066	67	1
2007-02-12 10:00:43.831	0.000	UDP	86.193.122.29:6881	->	147.251.192.170:61158	......	0	1	66	0	0	66	1
2007-02-12 10:00:43.832	0.000	UDP	147.251.192.170:61158	->	86.193.122.29:6881	......	0	1	104	0	0	104	1
2007-02-12 10:00:43.501	0.334	TCP	147.251.49.10:443	->	147.251.192.105:1404	.AP.SF	0	7	2551	20	61101	364	1
2007-02-12 10:00:43.499	0.343	TCP	147.251.192.105:1404	->	147.251.49.10:443	.AP.SF	0	7	1029	20	23999	147	1
2007-02-12 10:00:43.828	0.034	TCP	147.251.192.9:80	->	213.29.7.70:51071	.AP.SF	0	5	418	147	98352	83	1
2007-02-12 10:00:43.828	0.036	TCP	213.29.7.70:51071	->	147.251.192.9:80	.AP.SF	0	5	772	138	171555	154	1
2007-02-12 10:00:43.905	0.000	UDP	86.212.219.202:6881	->	147.251.192.170:61158	......	0	1	89	0	0	89	1
2007-02-12 10:00:43.908	0.000	UDP	147.251.192.170:61158	->	86.212.219.202:6881	......	0	1	234	0	0	234	1
2007-02-12 10:00:43.909	0.001	UDP	147.251.18.65:59861	->	147.251.195.131:137	......	0	2	148	1999	1.1 M	74	1
2007-02-12 10:00:43.910	0.004	UDP	147.251.195.131:137	->	147.251.18.65:59861	......	0	2	434	499	867999	217	1
2007-02-12 10:00:43.947	0.000	UDP	147.251.192.1:53859	->	85.17.42.212:53	......	0	1	73	0	0	73	1
2007-02-12 10:00:43.950	0.000	UDP	10.192.192.25:123	->	10.192.192.1:123	......	192	1	72	0	0	72	1
2007-02-12 10:00:43.951	0.000	UDP	192.42.93.30:53	->	147.251.192.1:53859	......	0	1	300	0	0	300	1
2007-02-12 10:00:43.975	0.000	UDP	85.17.42.212:53	->	147.251.192.1:53859	......	0	1	327	0	0	327	1
2007-02-12 10:00:43.969	0.006	UDP	147.251.18.65:59861	->	147.251.195.132:137	......	0	2	148	333	197333	74	1
2007-02-12 10:00:43.975	0.000	UDP	147.251.195.132:137	->	147.251.18.65:59861	......	0	2	470	0	0	235	1
2007-02-12 10:00:44.003	0.014	TCP	147.251.192.89:3059	->	81.95.96.121:80	.AP..F	0	5	484	357	276571	96	1
2007-02-12 10:00:44.003	0.018	TCP	81.95.96.121:80	->	147.251.192.89:3059	.AP.S.	0	5	2044	277	908444	408	1
2007-02-12 10:00:44.027	0.000	UDP	147.251.195.133:137	->	147.251.18.65:59861	......	0	2	470	0	0	235	1
2007-02-12 10:00:44.027	0.001	UDP	147.251.18.65:59861	->	147.251.195.133:137	......	0	2	148	1999	1.1 M	74	1
2007-02-12 10:00:44.026	0.010	TCP	147.251.192.89:3060	->	81.95.96.121:80	.AP.SF	0	5	595	499	475999	119	1
2007-02-12 10:00:44.026	0.021	TCP	81.95.96.121:80	->	147.251.192.89:3060	.AP.SF	0	5	338	238	128761	67	1
2007-02-12 10:00:39.140	4.942	TCP	86.41.152.176:49172	->	147.251.192.170:61158	.AP...	0	10	448	2	725	44	1
2007-02-12 10:00:44.087	0.000	UDP	147.251.18.65:59861	->	147.251.195.134:137	......	0	2	148	0	0	74	1
2007-02-12 10:00:43.975	0.112	TCP	205.188.165.249:80	->	147.251.193.4:1954	.AP.S.	0	3	646	26	46142	215	1
2007-02-12 10:00:44.728	0.000	UDP	147.251.195.134:137	->	147.251.18.65:59861	......	0	2	470	0	0	235	1

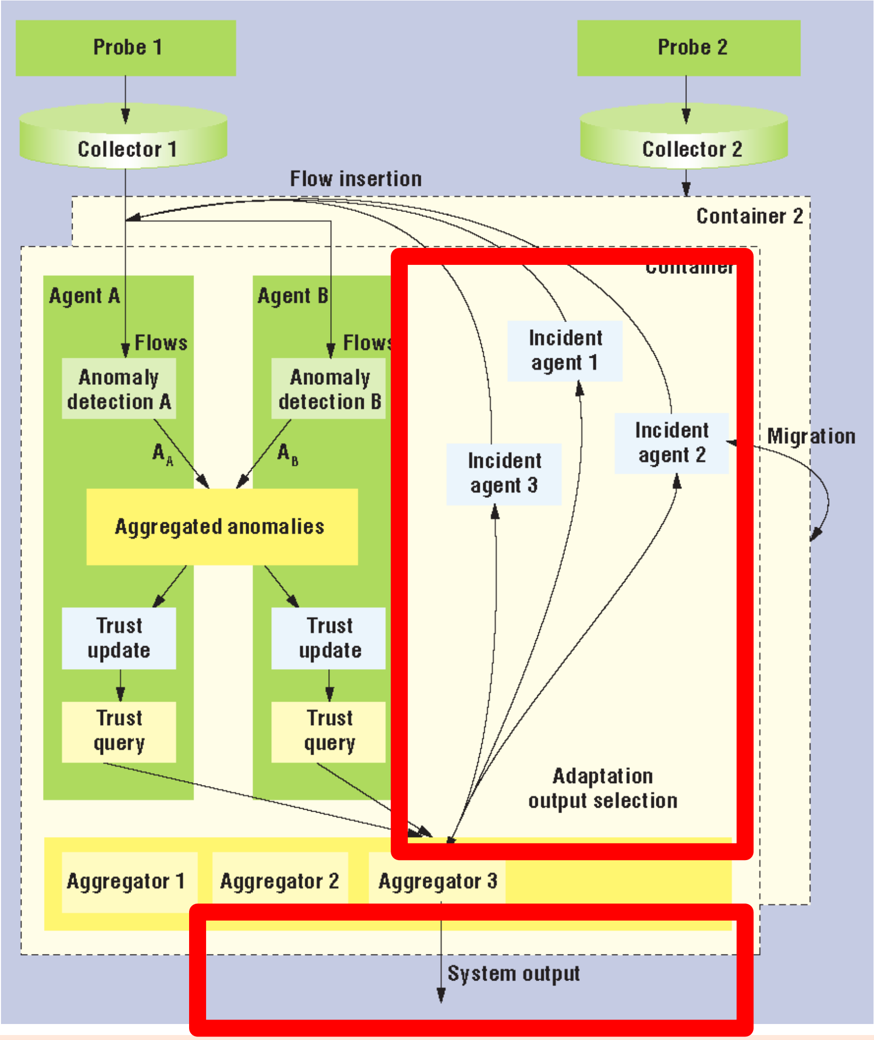
Detection Layer Overview (2)

Individual AD methods 300:2

Averaged anomalies 58:2

Averaged trust 15:2

Adaptive average 5:2



Anomaly Detection Methods

Method/Attack	Malware Brute force	Horizontal scanning	Vertical Sc. Fingerprint.	DoS/DDoS Flooding/Spoof.
MINDS	***	****	****	***
Xu	**	****	***	***
Xu-dst IP	*	*	**	*****
Lakhina - Volume	**	***	***	****
Lakhina - Entropy	***	****	**	***
TAPS	***	*****	*****	**

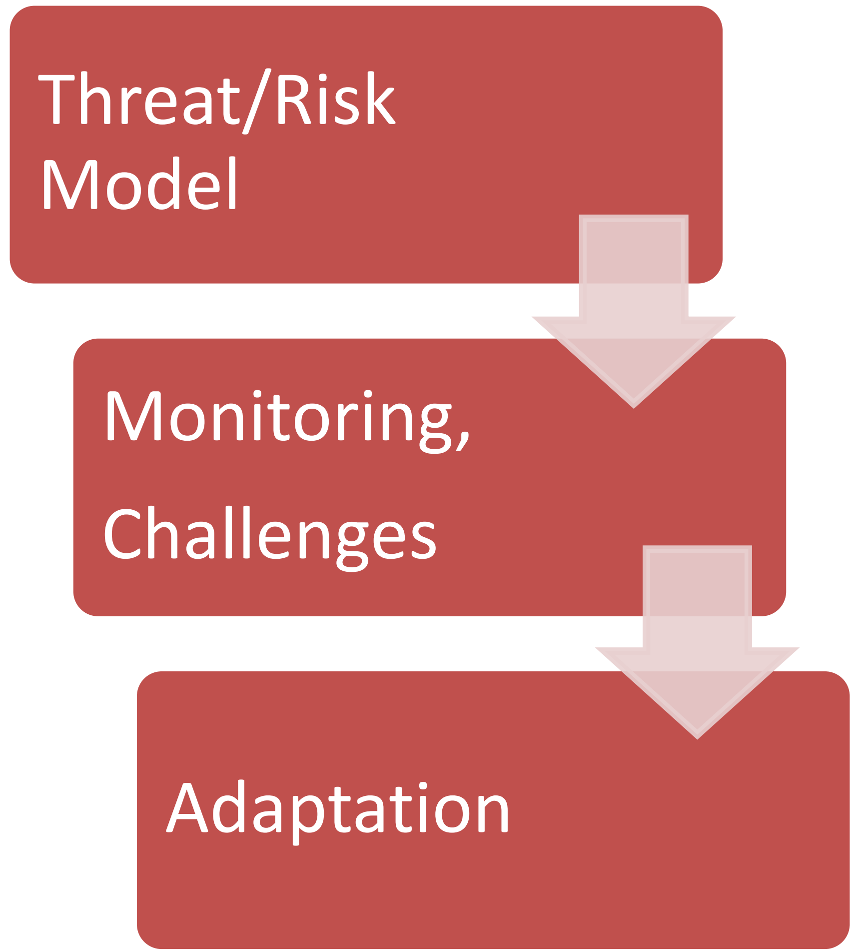
Reflective-Cognitive Principles

Cognition:

- Self-monitoring
- Self-evaluation
- Goal representation

Reflection:

- Component generation
- Component selection
- Component combination



Dynamic Classifier Selection

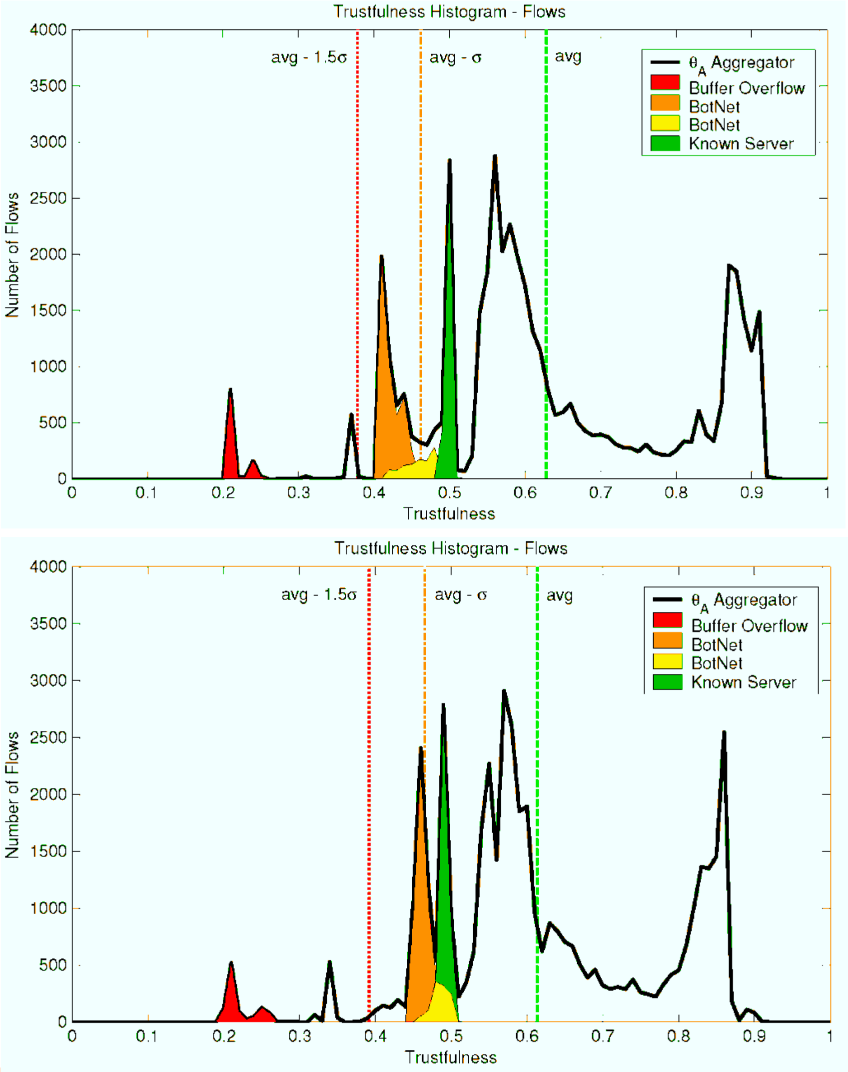
Unsupervised

Dynamic:

- Background traffic
- Model performance
- Attacks

Strategic behavior

- *Evasion*
- *Attacks on learning*



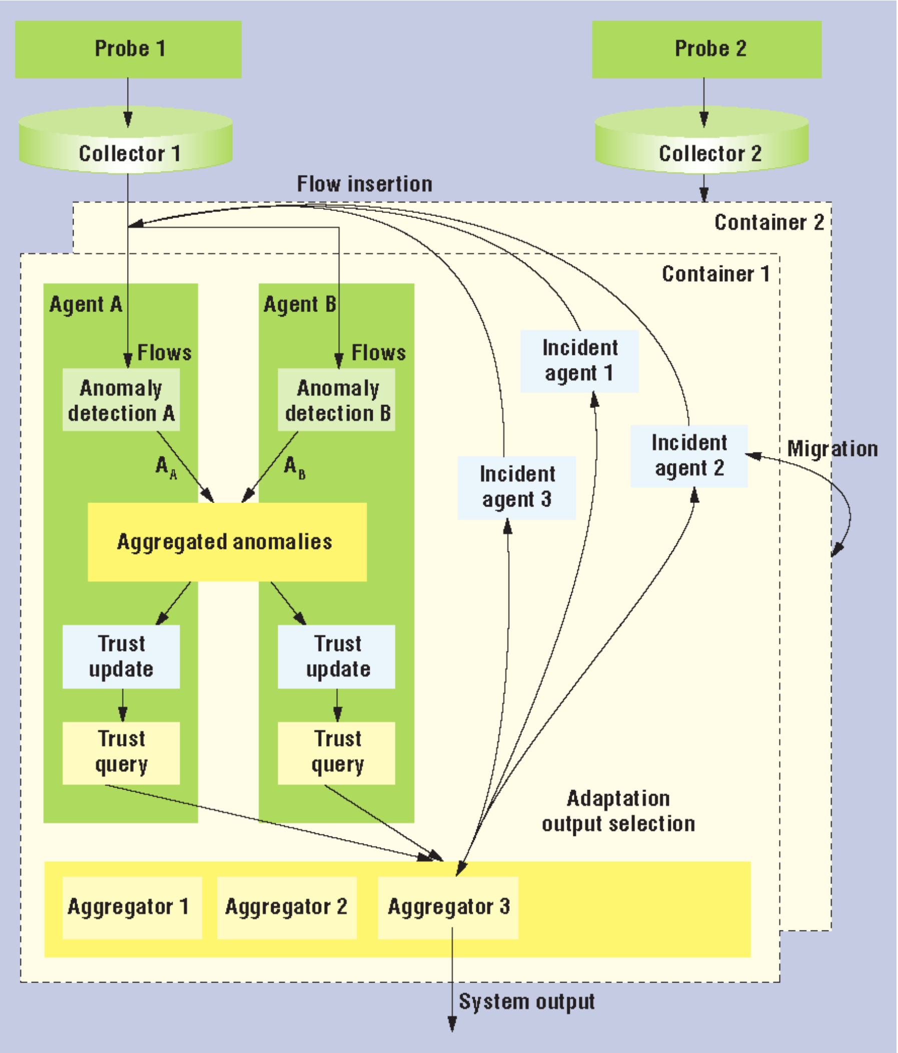
Adaptation Architecture

Monitoring & evaluation

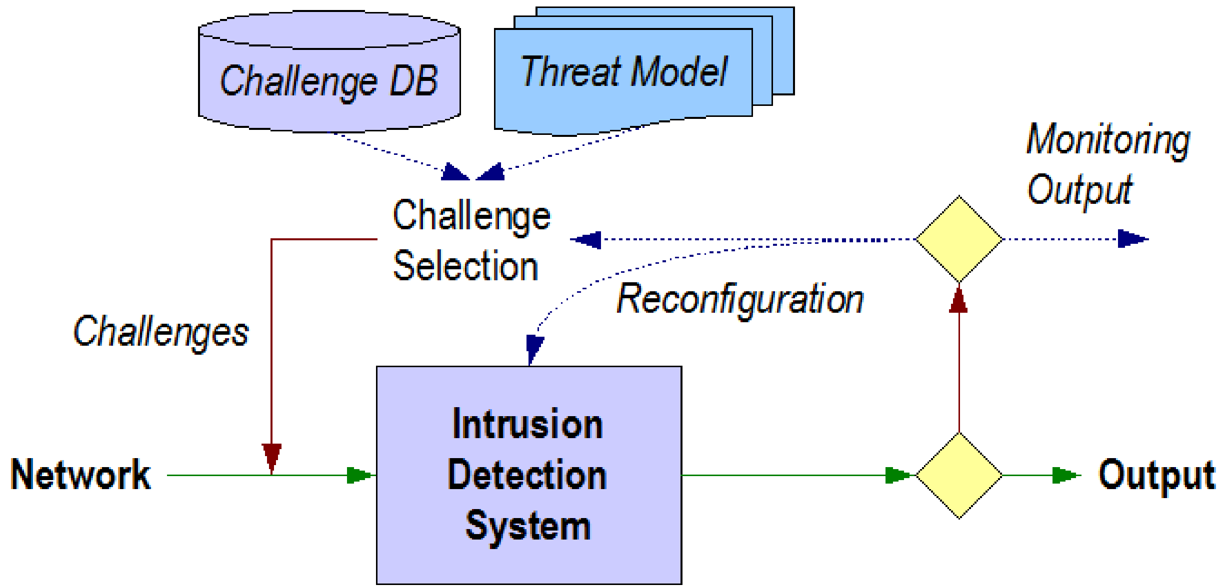
- Challenge insertion
- Challenge insertion control
- Challenge selection strategy

Adaptation

- Aggregation function selection
- Aggregation function creation



Monitoring: Challenge Insertion



Unlabeled background input data

Insertion of small set of challenges

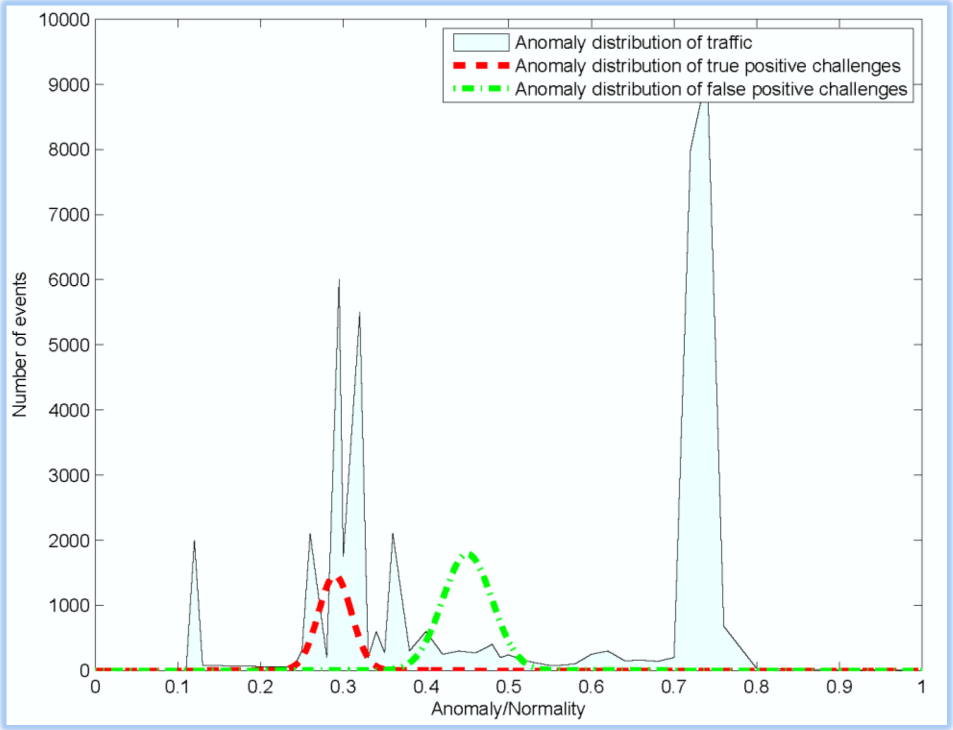
- Legitimate vs Malicious

From Challenge Insertion to Trust

Trust in the aggregator agent models its ability to separate the legitimate from malicious behavior under current conditions

Trust value used for:

- System monitoring
- Challenge control
- Self-adaptation



Trust Modeling – Issues

Regret/FIRE model individual reputation component used

- Startup delay considerations
- Changing network traffic character
- Number of inserted challenges vs. the number of attack types
- Relationship between the challenge insertion and trust

$$t_{\alpha}^{i,k} = \frac{\bar{y} - \bar{x}^k}{\sigma_y + \sigma_x^k}$$

$$T_{\alpha}^{k} = \sum_i w_i * t_{\alpha}^{i,k} \qquad w_i = \frac{1}{W} e^{(j-i)\frac{\ln(0.1)}{4}}$$

Evaluation

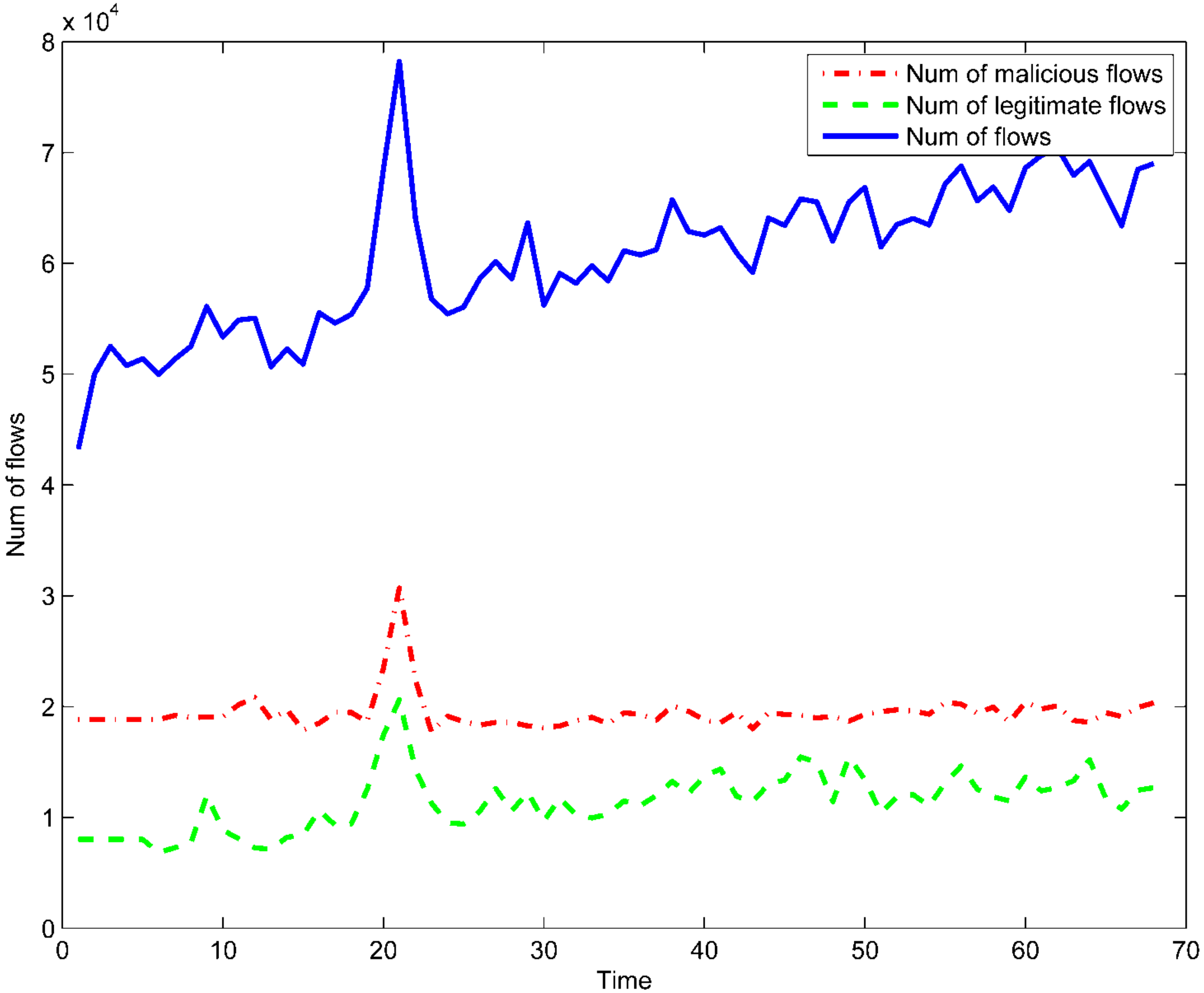
Real network traffic

- 1Gb link
- 200-300 Mb/sec eff.
- 200 flows/sec
- 6 hours … 70 datasets
- 5 minute collection

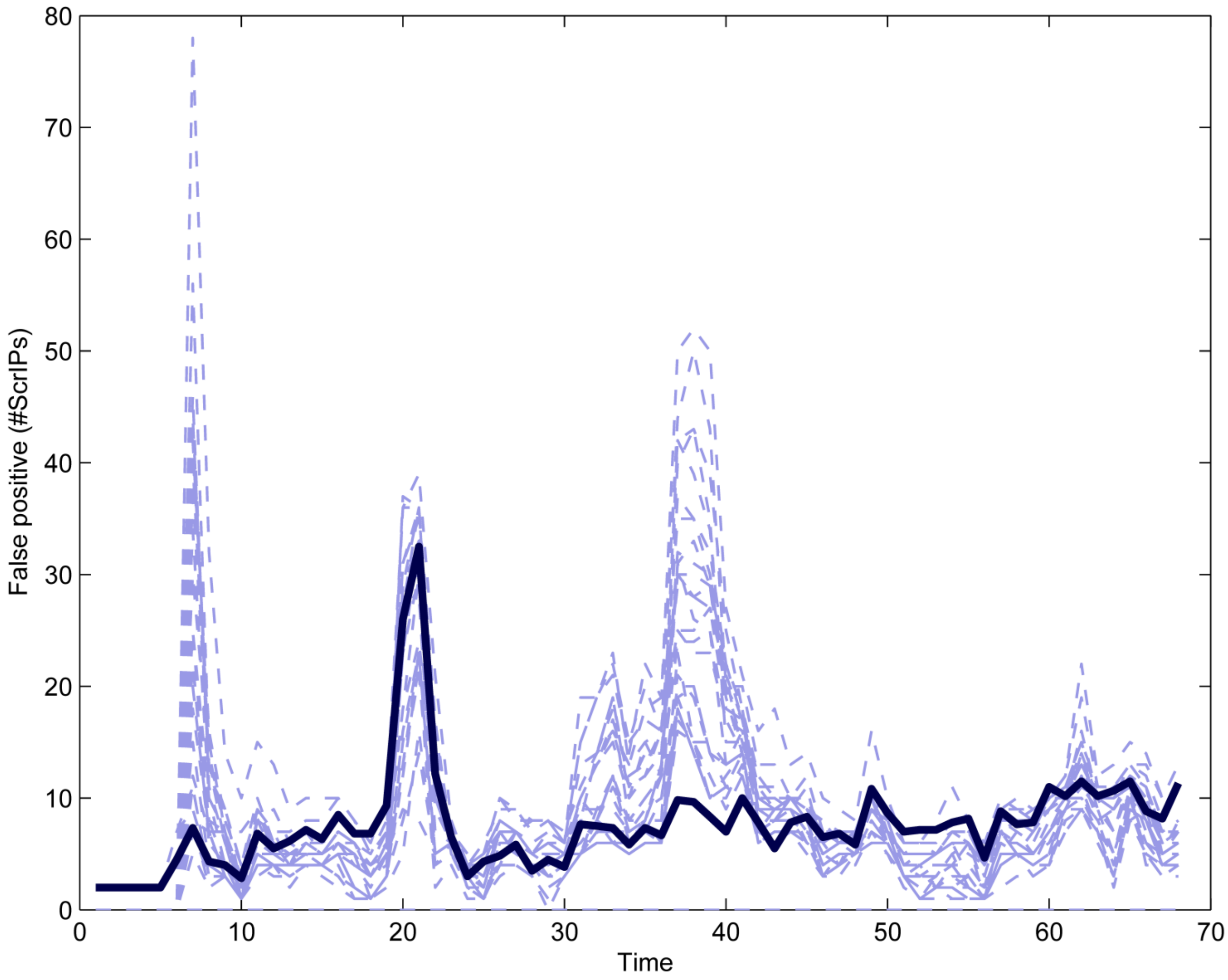
Third party attacks, manually classified

SSH scans, password brute force, worms/botnets, malware, P2P

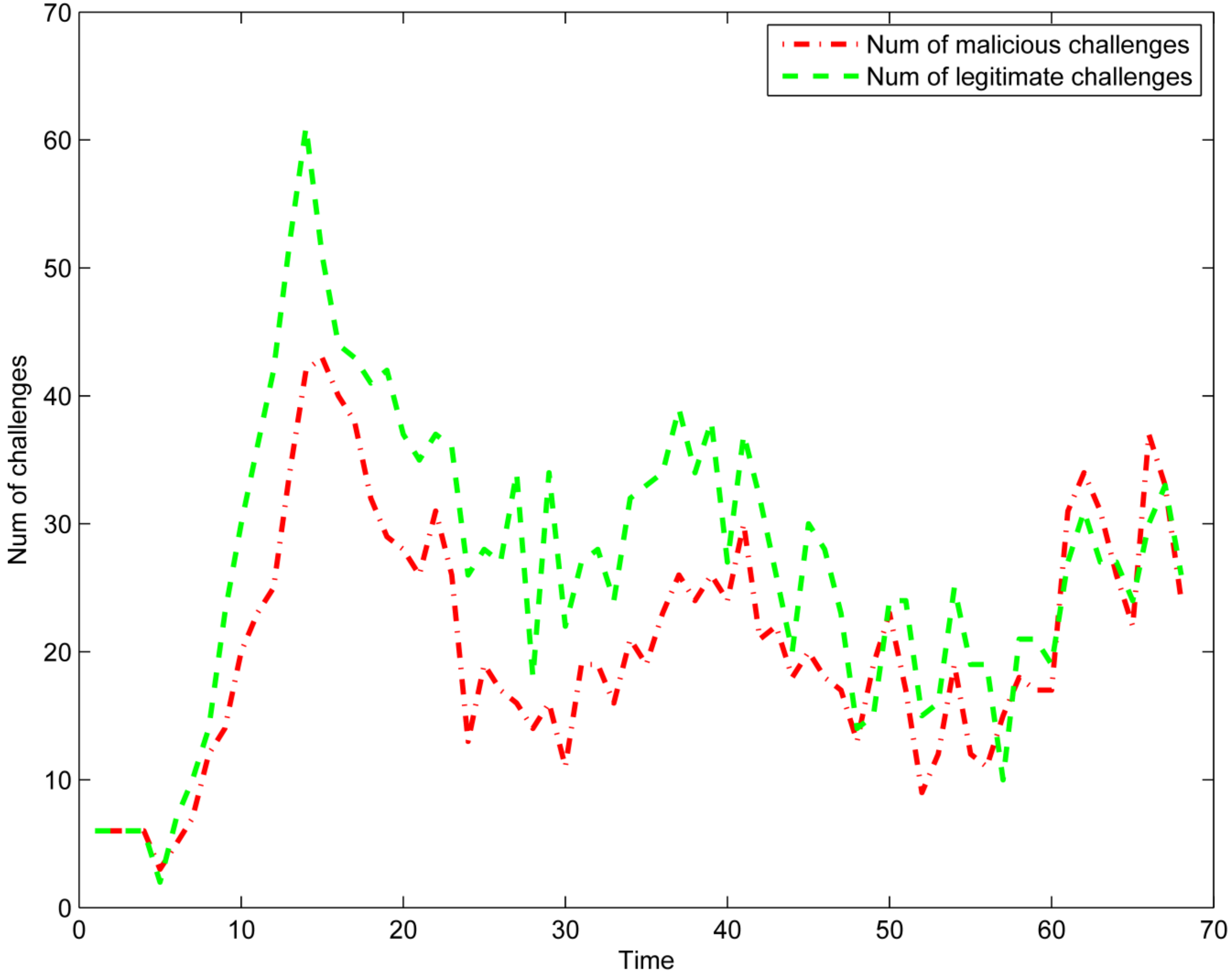
Experimental results



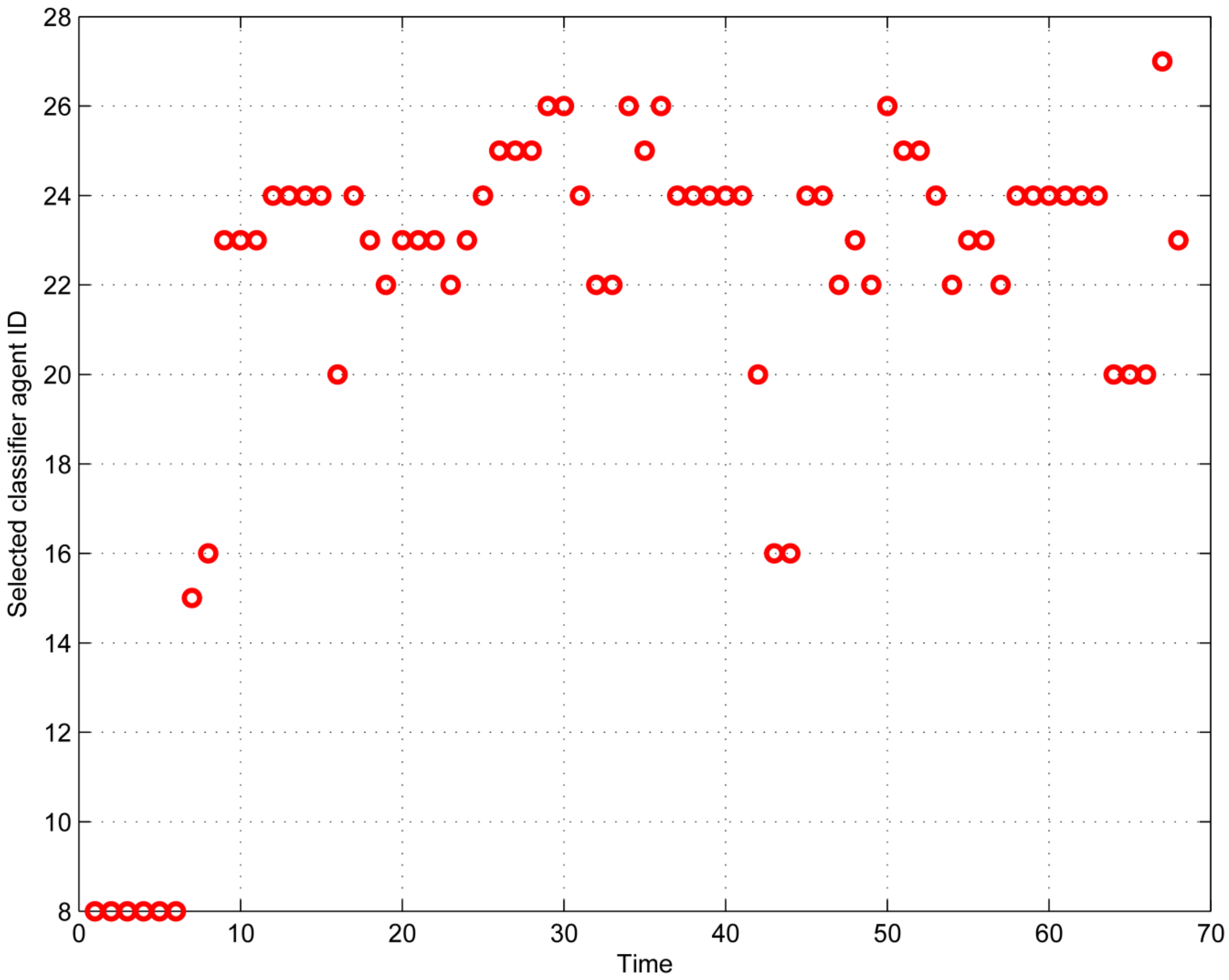
Experimental results (2)



Experimental results (3)



Experimental results (4)



Experimental results (5)

False positives reduced (excesses avoided)

False negatives comparable/reduced

Aggregation	False Negative (sIP)	False Positive (sIP)
Arithmetic average	14.7	12.5
Average aggregation fct.	13.1	24.3
Min FP aggregation fct.	14.5	5.3
Min FN aggregation fct.	**9.8**	125.2
Best aggregation fct.	13.7	5.7
Adaptive selection	**14.0**	**3.1**

- University network, third party attacks only – scans, P2P, password bf,…

Attack Modeling for Challenge Insertion

Trust as a **system monitoring** tool:

What is our IDS/NBA good for ?

Does it work right now ?

How sensitive it is ?

Can it detect X ?

Attack Trees

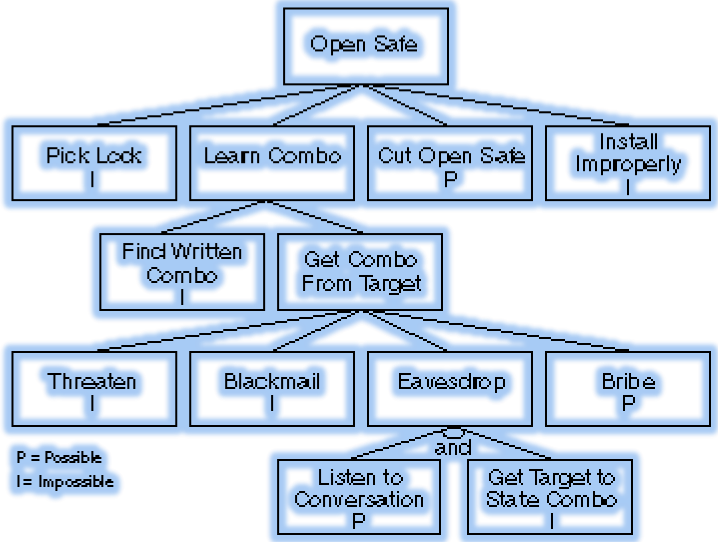
Model **attack options** to achieve a specific goal

Less expressive than plan representation techniques

- No action order
- AND and OR nodes

Intuitive

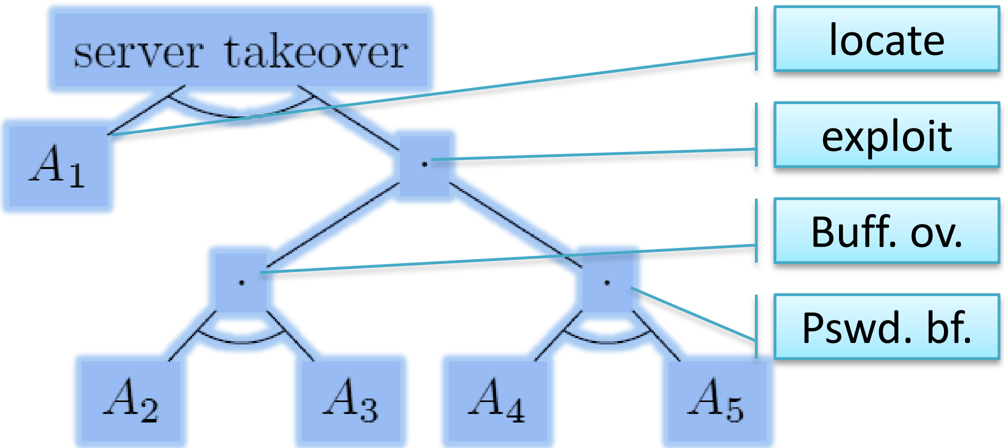
Typically quite large for real world problems



Schneier, 1999

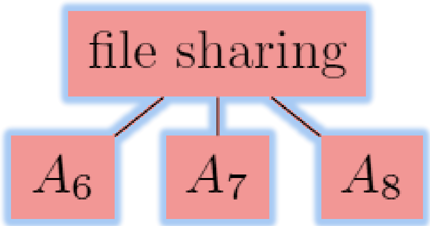
Attack Trees - Examples

Server take-over



A_1	horizontal scan
A_2	fingerprinting
A_3	buffer overflow
A_4	SSH brute force request
A_5	SSH brute force response

File sharing



A_6	download
A_7	upload
A_8	directory node

Decision-Theoretic Threat Modeling

One attack tree per threat (attacker objective)

Attack trees (roots) annotated with **loss values**

Loss values can be propagated to leaf nodes (i.e. attack actions)

Leaf nodes get loss value as well

Loss value can be determined for **attack classes**

$$F(T) = \{\{A_1\}, \{A_2, A_3\}, \{A_2, A_4\}\}$$

$$P(A_i, T_j) := \frac{1}{|F(T_j)|} \sum_{\substack{C_k \in F(T_j), \\ \text{with } A_i \in C_k}} \frac{1}{|C_k|}$$

$$P(A_i) := \frac{1}{\sum_{T_j \in T} D(T_j)} \cdot \sum_{T_k \in T} D(T_k) \cdot P(A_i, T_k)$$

Attack-Type Insertion Effects

Observable effects on trustfulness values for specific attack types

Slow, low volume attacks are still undetectable

So far inconclusive on the extracted event level

Attack	All challenges	Selected challenges
Horizontal scan	1.1/-0.2	1.4/0.0
Vertical scan	1.2/-0.2	1.4/0.3
Fingerprinting	1.5/1.2	1.9/1.6
SSH pass. brute force	-0.2/0.6	0.17/1.2
Buffer overflow	-0.2/0.1	0.2/0.0

Evasion and Strategic Behavior

Continuously, dynamically **optimized** IDS is better than **optimal** IDS.

Optimality is predictable.

Adaptation must be designed w.r.t. **intelligent**, **informed** and **strategically behaving** opponent

We assume **public knowledge** of NBA algorithms and trust model algorithms.

Attacker may **know/shape** other network traffic

Security should be achieved by **our strategic behavior**.

Game theory is an appropriate formalism.

Evasion: Two Time Horizons

Single interval

Defined by IDS observation interval

Attacker strategies:

- 0..n attacks actions, defined by type and parameters
- Resources used

Defender strategies:

- Detection agent set
- Aggregation function
- IDS parameters

Sequential

Sequence of intervals – 0..T

Attacker: attack plan execution

Plan unknown by defender

Defender: strategy sequence, minimizing the chance of generic attack plan success

Evasion: Two Time Horizons

Single interval

Defined by IDS observation interval

Attacker strategies:

- 0..n attacks actions, defined by type and parameters
- Resources used

Defender strategies:

- Detection agent set
- Aggregation function
- IDS parameters

Sequential

Sequence of intervals – 0..T

Attacker: attack plan execution

Plan unknown by defender

Defender: strategy sequence, minimizing the success of generic attack plan success

Game Definition

Our game definition is more realistic than other formulations

- Used for online definition, rather than analytical solution of an abstract problem

Normal form game is defined by

- **players**
- … their **strategies**
- their utility functions/**payoffs**

$$G = (P, S, U)$$

[P] Players (1)

Defender

Monitors **security policy**

Attack/Threat Models

Minimizes expected **undetected loss** and **administration costs**

- **False negatives**
- **False positives**

Attacker

Attacker **goals** & valuations

Needs to achieve undetected execution of an attack plan

Maximizes **attack gain**

Minimizes **detection probability**/penalty

[P] Players (2)

Defender

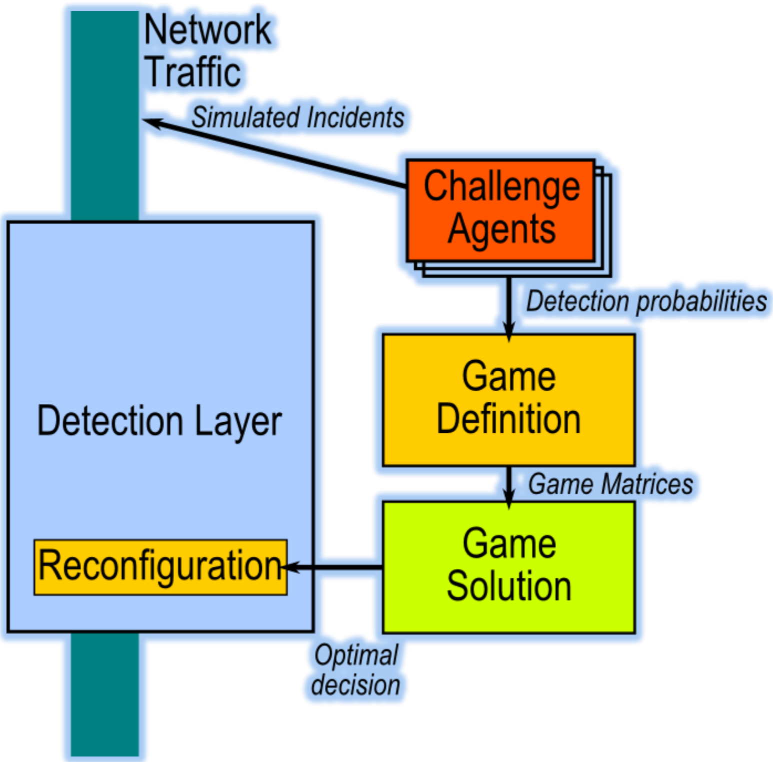
Monitors **security policy**

Attack/Threat Models

Minimizes expected **undetected loss** and **administration costs**

- **False negatives**
- **False positives**

Attacker Model



[S] Strategies

Defender

Aggregation function

- Determines sensitivity w.r.t. different attack types
- Wide range of behaviors
- Robustness
- Low predictability of individual strategies' response

Attacker

Attack type selection

- Game only models attack type: Horizontal scan, vertical scan, P2P sharing,
- Attack size/speed randomly drawn from a distribution
- Attack probabilities generated from the attack-tree based model (will change soon)

[S] Strategies - Defender

	$\alpha_{i,j}$ – P2P network	$\alpha_{i,j}$ – Horiz. scan	$\alpha_{i,j}$ – Vert. scan	β – Leg. traffic
Average	0.1979	0.3774	0.9441	0.3416
eowa1	0.1763	0.2672	0.9236	0.3568
eowa2	0.1709	0.2672	0.9236	0.3459
eowa3	0.1925	0.3774	0.9236	0.3480
eowa4	0.1817	0.3774	0.9133	0.3848
eowa5	0.0818	0.2500	0.9571	0.3222
eowa6	0.1871	0.2672	0.9236	0.3440
eowa7	0.1709	0.3774	0.9236	0.3599
eowa8	0.1925	0.3774	0.9236	0.3080
eowa9	0.1421	0.2500	1.0000	0.3224
owa1	0.1763	0.4054	0.9133	0.3633
owa10	0.1789	0.3774	0.8704	0.4515
owa2	0.1709	0.2672	0.8986	0.3300
owa3	0.1763	0.4054	0.9133	0.3402
owa4	0.0818	0.2500	0.9571	0.3224
owa5	0.1709	0.2672	0.9089	0.3771
owa6	0.1752	0.2500	0.9571	0.3384
owa7	0.1817	0.2672	0.9133	0.3342
owa8	0.1817	0.2672	0.8704	0.4211
owa9	0.1911	0.3774	0.9441	0.4526
wavg2	0.2071	0.4054	0.7227	0.4395
wavg3	0.3288	0.2500	0.9571	0.3184

[S] Strategies - Attacker

Attack class		γ_j	$P_a(a_j)$	$P_d(a_j)$
Apache benchmark	a_1	0.001	300	-300
Horizontal scan	a_2	0.001	140	-140
P2P Network	a_3	0.001	180	-180
SSH brute force request	a_4	0.001	1000	-1000
SSH brute force response	a_5	0.001	1000	-1000
Vertical scan	a_6	0.001	150	-150

[G] Normal Form Game

Defender/Attacker [i/j]	Strategy 1 (Attack class$_1$)	Strategy 2 (Attack class$_2$)	Strategy 3 (Attack class$_3$)
Strategy 1 (Aggregation operator$_1$)	…	…	…
Strategy 1 (Aggregation operator$_2$)	…	**Payoffs** Defender and Attacker	…
Strategy 1 (Aggregation operator$_3$)	…	…	…

[U] Utility Functions - Measurements

Values depending on the environment, not under the direct control of the players

- Detection probability
- Attack success probability
- False positive probability
- Background traffic volume

[U] Utility Functions – Costs/Payoffs

- Coefficients representing players’ preferences, policies and utilities

Defender	**Attacker**
Attack success payoff/loss	Attack success payoff
Attack detection payoff/loss	Detection payoff/loss
TP processing cost	Attack cost
FP processing cost	
Monitoring fixed cost	

[U] Defender

$$u_d(d_j, a_i, t) = \overbrace{\alpha_{i,j}(D_d(a_j) - C_{TP})}^{\text{Detected incidents}} + \overbrace{(1 - \alpha_{i,j})\gamma_j P_d(a_j)}^{\text{Undetected incidents}} - \overbrace{\beta V(t) C_{FP} - C_M}^{\text{Monitoring}}$$

[U] Defender - Detected

$$u_d(d_j, a_i, t) = \overbrace{\alpha_{i,j}(D_d(a_j) - C_{TP})}^{\text{Detected incidents}} + \overbrace{(1-\alpha_{i,j})\gamma_j P_d(a_j)}^{\text{Undetected incidents}} - \overbrace{\beta V(t) C_{FP} - C_M}^{\text{Monitoring}}$$

Detection probability
For attack *j* and
aggregation function *i*

Cost of true positive
handling

$$\alpha_{i,j}\underbrace{(D_d(a_j)}_{} - C_{TP})$$

Detection payoff
When an attack action
a_j is detected

[U] Defender - Undetected

$$u_d(d_j, a_i, t) = \underbrace{\alpha_{i,j}(D_d(a_j) - C_{TP})}_{\text{Detected incidents}} + \underbrace{(1-\alpha_{i,j})\gamma_j P_d(a_j)}_{\text{Undetected incidents}} - \underbrace{\beta V(t) C_{FP} - C_M}_{\text{Monitoring}}$$

Detection probability
For attack *j* and
aggregation function *i*

$$(1-\alpha_{i,j})\gamma_j P_d(a_j)$$

Attack a_j success
probability

Payoff (loss)
Cost of successful
undetected a_j

[U] Defender - Monitoring

$$u_d(d_j, a_i, t) = \overbrace{\alpha_{i,j}(D_d(a_j) - C_{TP})}^{\text{Detected incidents}} + \overbrace{(1 - \alpha_{i,j})\gamma_j P_d(a_j)}^{\text{Undetected incidents}} - \overbrace{\beta V(t) C_{FP} - C_M}^{\text{Monitoring}}$$

Traffic volume in number of flows

FP probability for each flow

Average cost per FP

$$-\underbrace{\beta V(t) C_{FP}}_{\text{Cost of FP processing}} - C_M$$

Fixed cost of monitoring

[U] Attacker

$$u_a(d_j, a_i) = \underbrace{\alpha_{i,j} D_a(a_j)}_{\text{Detected incidents}} + \underbrace{(1 - \alpha_{i,j})\gamma_j P_a(a_j)}_{\text{Undetected incidents}} - \underbrace{C_a(a_j)}_{\text{Attack cost}}$$

[U] Attacker - Detected

$$u_a(d_j, a_i) = \underbrace{\alpha_{i,j} D_a(a_j)}_{\text{Detected incidents}} + \underbrace{(1 - \alpha_{i,j})\gamma_j P_a(a_j)}_{\text{Undetected incidents}} - \underbrace{C_a(a_j)}_{\text{Attack cost}}$$

Detection probability For attack *j* and aggregation function *i*

Payoff/loss of the detection

$$\alpha_{i,j} D_a(a_j)$$

[U] Attacker - Undetected

$$u_a(d_j, a_i) = \overbrace{\alpha_{i,j} D_a(a_j)}^{\text{Detected incidents}} + \overbrace{(1 - \alpha_{i,j})\gamma_j P_a(a_j)}^{\text{Undetected incidents}} - \overbrace{C_a(a_j)}^{\text{Attack cost}}$$

Detection probability For attack *j* and aggregation function *i*

Payoff for the successful attack

$$(1 - \alpha_{i,j})\gamma_j P_a(a_j)$$

Success probability for the attack a_j

[U] Attacker – Attack Cost

$$u_a(d_j, a_i) = \underbrace{\alpha_{i,j} D_a(a_j)}_{\text{Detected incidents}} + \underbrace{(1 - \alpha_{i,j})\gamma_j P_a(a_j)}_{\text{Undetected incidents}} - \underbrace{C_a(a_j)}_{\text{Attack cost}}$$

$$- C_a(a_j)$$

Payoff for the successful attack a_j

[G] Situations - Detected

Detection payoff
When an attack action a_j is detected

Cost of true positive handling

$$\alpha_{i,j}(D_d(a_j) - C_{TP})$$

Detection probability
For attack *j* and aggregation function *i*

Payoff/loss of the detection

$$\alpha_{i,j} D_a(a_j)$$

- **Outcomes**: Da,Dd ~ 0 on the Internet/first layers of defense
- Increases with network value and decreasing attack frequency
- TP handling cost alone makes IDS on the gateways useful mainly for attack intelligence/automated statistical processing

[G] Situations - Undetected

Detection payoff
When an attack action a_j is detected

Payoff/loss from the successful undetected breach

$$(1 - \alpha_{i,j})\gamma_j P_d(a_j)$$

Detection probability
For attack a_j and aggregation function *i*

Attack a_j success probability

$$(1 - \alpha_{i,j})\gamma_j P_a(a_j)$$

Payoff for the successful undetected breach

- **Outcomes**: Success probability amortizes payoffs
- Typically $P_d > P_a$, $P_a > D_a$, and $P_d > D_d$ – monitoring is essential
- Best outcome for the defender: undetected, ineffective attack

Utilities in Game

Attacker

$u_a =$

Def./Att.	a_1	a_2	a_3	a_4	a_5	a_6
Average	−0.11	0.03	0.11	−0.72	−0.45	−0.13
eowa1	−0.13	0.07	0.12	−0.70	−0.43	−0.13
eowa2	−0.13	0.07	0.12	−0.70	−0.43	−0.13
eowa3	−0.02	0.03	0.11	−0.68	−0.34	−0.13
eowa4	0.05	0.03	0.11	−0.66	−0.17	−0.12
eowa5	0.06	0.07	0.15	−0.41	−0.47	−0.14
eowa6	−0.11	0.07	0.11	−0.70	−0.43	−0.13
eowa7	−0.20	0.03	0.12	−0.72	−0.17	−0.13
eowa8	−0.02	0.03	0.11	−0.70	−0.23	−0.13
eowa9	−0.06	0.07	0.13	−0.41	0.00	−0.15
owa1	0.05	0.03	0.12	−0.66	−0.16	−0.12
owa10	−0.08	0.03	0.12	−0.44	−0.10	−0.11
owa2	−0.13	0.07	0.12	−0.66	−0.31	−0.12
owa3	0.05	0.03	0.12	−0.66	−0.16	−0.12
owa4	0.06	0.07	0.15	−0.41	−0.47	−0.14
owa5	−0.13	0.07	0.12	−0.68	−0.44	−0.12
owa6	0.06	0.07	0.12	−0.41	−0.62	−0.14
owa7	0.01	0.07	0.11	−0.66	−0.50	−0.12
owa8	−0.13	0.07	0.11	−0.44	−0.43	−0.11
owa9	0.14	0.03	0.11	−0.86	0.15	−0.13
wavg2	−0.02	0.03	0.11	−0.75	0.17	−0.07
wavg3	0.00	0.07	0.06	−0.41	−0.62	−0.14

Defender

Def./Att.	a_1	a_2	a_3	a_4	a_5	a_6
Average	−0.56	−0.70	−0.77	0.05	−0.21	−0.53
eowa1	−0.55	−0.74	−0.79	0.02	−0.25	−0.55
eowa2	−0.54	−0.74	−0.79	0.03	−0.24	−0.54
eowa3	−0.64	−0.69	−0.77	0.02	−0.32	−0.53
eowa4	−0.71	−0.69	−0.77	0.01	−0.49	−0.53
eowa5	−0.62	−0.63	−0.71	−0.15	−0.09	−0.42
eowa6	−0.57	−0.74	−0.79	0.02	−0.25	−0.55
eowa7	−0.45	−0.67	−0.76	0.08	−0.47	−0.51
eowa8	−0.63	−0.68	−0.75	0.06	−0.42	−0.52
eowa9	−0.52	−0.65	−0.71	−0.17	−0.58	−0.43
owa1	−0.71	−0.68	−0.77	0.00	−0.50	−0.53
owa10	−0.55	−0.66	−0.74	−0.19	−0.52	−0.51
owa2	−0.53	−0.72	−0.77	0.01	−0.34	−0.53
owa3	−0.70	−0.68	−0.77	0.01	−0.49	−0.53
owa4	−0.61	−0.62	−0.70	−0.14	−0.08	−0.41
owa5	−0.55	−0.74	−0.80	0.00	−0.24	−0.56
owa6	−0.74	−0.75	−0.80	−0.27	−0.06	−0.54
owa7	−0.70	−0.76	−0.81	−0.03	−0.19	−0.57
owa8	−0.51	−0.70	−0.75	−0.20	−0.21	−0.53
owa9	−0.69	−0.58	−0.66	0.31	−0.70	−0.42
wavg2	−0.60	−0.64	−0.72	0.14	−0.79	−0.55
wavg3	−0.67	−0.74	−0.73	−0.26	−0.05	−0.53

[G] Solution Concepts

Dominant strategy: best strategy against any strategy of the opponent

- Exists only rarely
- Easy to guess

Max-Min: strategy that guarantees minimal worst-case damage

- Robust and easy to find
- May be too pessimistic
- Provides best results against insiders

[G] Solution Concepts (2)

Conditional dominance: iteratively removes dominated strategies, then selects randomly

- Unstable, used for benchmark only
- Represents the behavior that is not naïve, but is not strategical

Nash equilibrium: couples of stable strategies where unilateral change is unprofitable

- Robust, but implies game knowledge/learning
- Guaranteed to exist, but difficult to find

Solutions: Dominant Strategy

Does not exist in this game

- We have experimentally found that it exists only in about 1 game in 20 for our system, traffic and settings ranges
- It’s existence in the example would make all other solutions trivial

Solutions: Max-Min

Exists for all games

- Easy to determine
- Unilateral – opponent model knowledge not necessary

$u_d =$

Def./Att.	a_1	a_2	a_3	a_4	a_5	a_6
Average	−0.56	−0.70	−0.77	0.05	−0.21	−0.53
*eowa*1	−0.55	−0.74	−0.79	0.02	−0.25	−0.55
*eowa*2	−0.54	−0.74	−0.79	0.03	−0.24	−0.54
*eowa*3	−0.64	−0.69	−0.77	0.02	−0.32	−0.53
*eowa*4	−0.71	−0.69	−0.77	0.01	−0.49	−0.53
*eowa*5	−0.62	−0.63	−0.71	−0.15	−0.09	−0.42
*eowa*6	−0.57	−0.74	−0.79	0.02	−0.25	−0.55
*eowa*7	−0.45	−0.67	−0.76	0.08	−0.47	−0.51
*eowa*8	−0.63	−0.68	−0.75	0.06	−0.42	−0.52
*eowa*9	−0.52	−0.65	−0.71	−0.17	−0.58	−0.43
*owa*1	−0.71	−0.68	−0.77	0.00	−0.50	−0.53
*owa*10	−0.55	−0.66	−0.74	−0.19	−0.52	−0.51
*owa*2	−0.53	−0.72	−0.77	0.01	−0.34	−0.53
*owa*3	−0.70	−0.68	−0.77	0.01	−0.49	−0.53
*owa*4	−0.61	−0.62	−0.70	−0.14	−0.08	−0.41
*owa*5	−0.55	−0.74	−0.80	0.00	−0.24	−0.56
*owa*6	−0.74	−0.75	−0.80	−0.27	−0.06	−0.54
*owa*7	−0.70	−0.76	−0.81	−0.03	−0.19	−0.57
*owa*8	−0.51	−0.70	−0.75	−0.20	−0.21	−0.53
*owa*9	−0.69	−0.58	−0.66	0.31	−0.70	−0.42
*wavg*2	−0.60	−0.64	−0.72	0.14	−0.79	−0.55
*wavg*3	−0.67	−0.74	−0.73	−0.26	−0.05	−0.53

Solutions: Conditional Dominance

- Selects a wide set of undominated strategies
- Randomly picks one strategy from the set
- Only a benchmark solution, not a real concept

$u_d =$

$Def./Att.$	a_1	a_2	a_3	a_4	a_5	a_6
$Average$	-0.56	-0.70	-0.77	0.05	-0.21	-0.53
$eowa1$	-0.55	-0.74	-0.79	0.02	-0.25	-0.55
$eowa2$	-0.54	-0.74	-0.79	0.03	-0.24	-0.54
$eowa3$	-0.64	-0.69	-0.77	0.02	-0.32	-0.53
$eowa4$	-0.71	-0.69	-0.77	0.01	-0.49	-0.53
$eowa5$	-0.62	-0.63	-0.71	-0.15	-0.09	-0.42
$eowa6$	-0.57	-0.74	-0.79	0.02	-0.25	-0.55
$eowa7$	-0.45	-0.67	-0.76	0.08	-0.47	-0.51
$\boxed{eowa8}$	-0.63	-0.68	-0.75	0.06	-0.42	-0.52
$eowa9$	-0.52	-0.65	-0.71	-0.17	-0.58	-0.43
$owa1$	-0.71	-0.68	-0.77	0.00	-0.50	-0.53
$owa10$	-0.55	-0.66	-0.74	-0.19	-0.52	-0.51
$owa2$	-0.53	-0.72	-0.77	0.01	-0.34	-0.53
$owa3$	-0.70	-0.68	-0.77	0.01	-0.49	-0.53
$owa4$	-0.61	-0.62	-0.70	-0.14	-0.08	-0.41
$owa5$	-0.55	-0.74	-0.80	0.00	-0.24	-0.56
$owa6$	-0.74	-0.75	-0.80	-0.27	-0.06	-0.54
$owa7$	-0.70	-0.76	-0.81	-0.03	-0.19	-0.57
$owa8$	-0.51	-0.70	-0.75	-0.20	-0.21	-0.53
$owa9$	-0.69	-0.58	-0.66	0.31	-0.70	-0.42
$wavg2$	-0.60	-0.64	-0.72	0.14	-0.79	-0.55
$wavg3$	-0.67	-0.74	-0.73	-0.26	-0.05	-0.53

Solutions: Nash Equilibria

NE exists for any “reasonable” game

Game can have several equilibria

Some equilibria are mixed

Two sided concept

Players need game matrix knowledge or learning time to find it

	Def./Att.	a_1	a_2	a_3	a_4	a_5	a_6
	Average	−0.56	−0.70	−0.77	0.05	−0.21	−0.53
	eowa1	−0.55	−0.74	−0.79	0.02	−0.25	−0.55
	eowa2	−0.54	−0.74	−0.79	0.03	−0.24	−0.54
	eowa3	−0.64	−0.69	−0.77	0.02	−0.32	−0.53
	eowa4	−0.71	−0.69	−0.77	0.01	−0.49	−0.53
	eowa5	−0.62	−0.63	−0.71	−0.15	−0.09	−0.42
	eowa6	−0.57	−0.74	−0.79	0.02	−0.25	−0.55
	eowa7	−0.45	−0.67	−0.76	0.08	−0.47	−0.51
	eowa8	−0.63	−0.68	−0.75	0.06	−0.42	−0.52
0.11	eowa9	−0.52	−0.65	−0.71	−0.17	−0.58	−0.43
$u_d =$	owa1	−0.71	−0.68	−0.77	0.00	−0.50	−0.53
	owa10	−0.55	−0.66	−0.74	−0.19	−0.52	−0.51
	owa2	−0.53	−0.72	−0.77	0.01	−0.34	−0.53
	owa3	−0.70	−0.68	−0.77	0.01	−0.49	−0.53
0.04	owa4	−0.61	−0.62	−0.70	−0.14	−0.08	−0.41
	owa5	−0.55	−0.74	−0.80	0.00	−0.24	−0.56
	owa6	−0.74	−0.75	−0.80	−0.27	−0.06	−0.54
	owa7	−0.70	−0.76	−0.81	−0.03	−0.19	−0.57
	owa8	−0.51	−0.70	−0.75	−0.20	−0.21	−0.53
0.85	owa9	−0.69	−0.58	−0.66	0.31	−0.70	−0.42
	wavg2	−0.60	−0.64	−0.72	0.14	−0.79	−0.55
	wavg3	−0.67	−0.74	−0.73	−0.26	−0.05	−0.53

Solutions: Nash Equilibria (2)

- NE exists for any “reasonable” game
- Game can have several equilibria
- Some equilibria are mixed
- Two sided concept
- Players need game matrix knowledge or learning time to find it

$u_a =$

	0.21		*0.77*	*0.02*		
Def./Att.	a_1	a_2	a_3	a_4	a_5	a_6
Average	−0.11	0.03	0.11	−0.72	−0.45	−0.13
eowa1	−0.13	0.07	0.12	−0.70	−0.43	−0.13
eowa2	−0.13	0.07	0.12	−0.70	−0.43	−0.13
eowa3	−0.02	0.03	0.11	−0.68	−0.34	−0.13
eowa4	0.05	0.03	0.11	−0.66	−0.17	−0.12
eowa5	0.06	0.07	0.15	−0.41	−0.47	−0.14
eowa6	−0.11	0.07	0.11	−0.70	−0.43	−0.13
eowa7	−0.20	0.03	0.12	−0.72	−0.17	−0.13
eowa8	−0.02	0.03	0.11	−0.70	−0.23	−0.13
eowa9	−0.06	0.07	0.13	−0.41	0.00	−0.15
owa1	0.05	0.03	0.12	−0.66	−0.16	−0.12
owa10	−0.08	0.03	0.12	−0.44	−0.10	−0.11
owa2	−0.13	0.07	0.12	−0.66	−0.31	−0.12
owa3	0.05	0.03	0.12	−0.66	−0.16	−0.12
owa4	0.06	0.07	0.15	−0.41	−0.47	−0.14
owa5	−0.13	0.07	0.12	−0.68	−0.44	−0.12
owa6	0.06	0.07	0.12	−0.41	−0.62	−0.14
owa7	0.01	0.07	0.11	−0.66	−0.50	−0.12
owa8	−0.13	0.07	0.11	−0.44	−0.43	−0.11
owa9	0.14	0.03	0.11	−0.86	0.15	−0.13
wavg2	−0.02	0.03	0.11	−0.75	0.17	−0.07
wavg3	0.00	0.07	0.06	−0.41	−0.62	−0.14

Experimental Evaluation

Specific attack scenario really executed on the background of real network traffic

Several executions of the plan with varying parameters

Attacks:

- Horizontal scan
- Vertical scan
- SSH brute force

Two settings with different defender's preferences

All results averaged from 40 runs with different challenges, same data

	Settings 1		Settings 2	
Attack class	P_d	γ_j	P_d	γ_j
Apache benchmark	300	0.001	600	0.001
Horizontal scan	140	0.001	300	0.001
P2P Network	180	0.001	300	0.001
SSH brute force request	1000	0.001	500	0.001
SSH brute force response	1000	0.001	500	0.001
Vertical scan	150	0.001	300	0.001

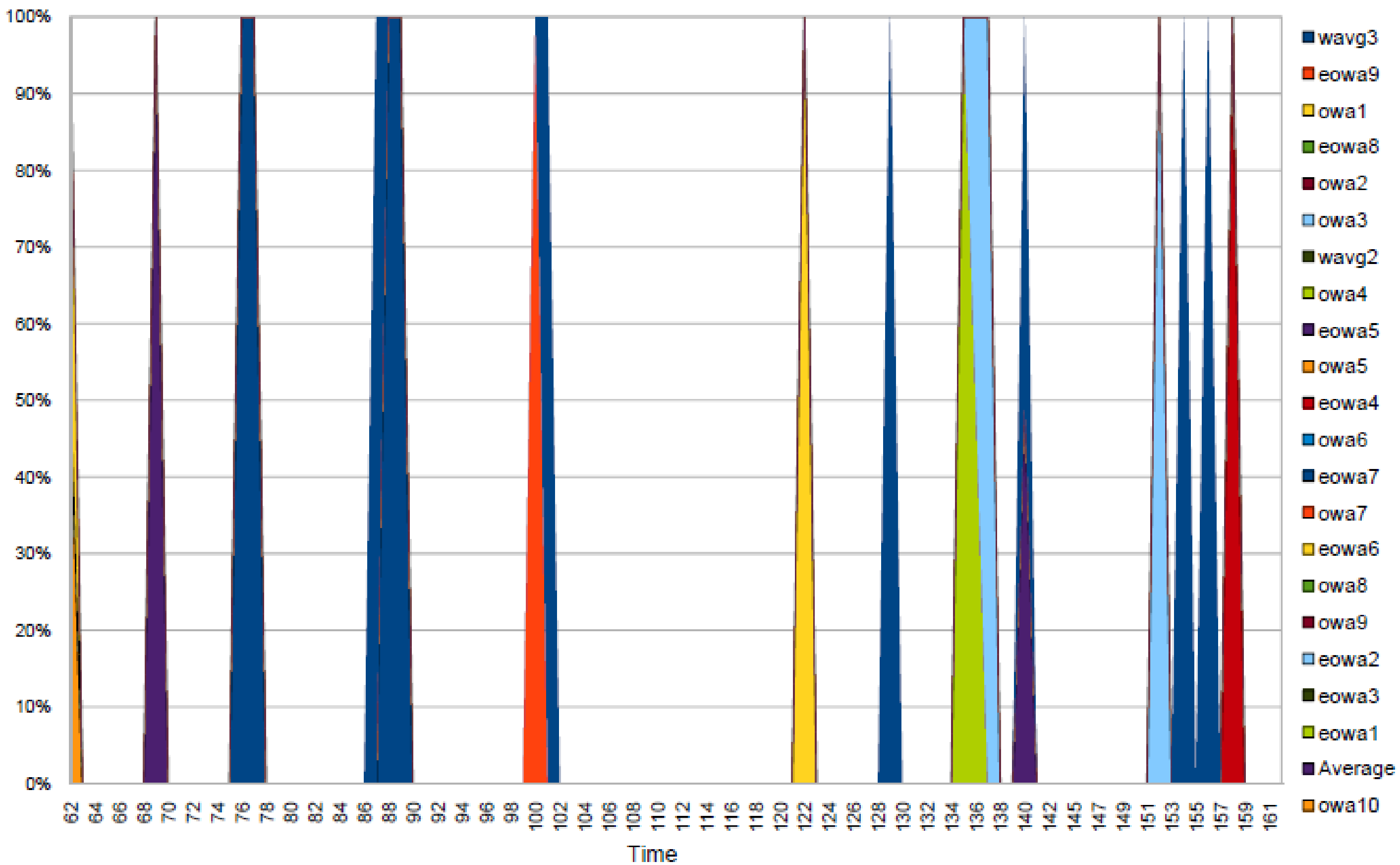
Experimental Evaluation

Detection payoffs against attack variants; settings 1, False positives included

Attack class	Strategies – settings 1						
	DS	CD	MM	NE	Trust	Best OWA	Avg. OWA
SSH brute force	0.04	7.27	21.38	16.41	17.61	136.69	9.08
Vertical TCP scan with OS detection	0.04	10.72	27.68	20.38	11.91	74.65	6.90
Vertical UDP scan	0.00	17.03	57.26	54.84	46.99	170.15	10.91
Horizontal TCP scan for SSH service	0.00	21.57	92.32	84.51	48.85	166.82	12.51
Horizontal UDP scan for DNS service	0.00	21.71	114.92	107.41	60.94	219.31	15.90
Horizontal ICMP ping scan	-0.21	23.08	96.13	76.50	56.08	169.55	11.19
Horizontal TCP scan for all services	0.68	16.98	66.50	57.09	35.04	118.87	13.23
Average	**0.08**	**16.91**	**68.03**	**59.66**	**39.63**	**150.86**	**11.39**

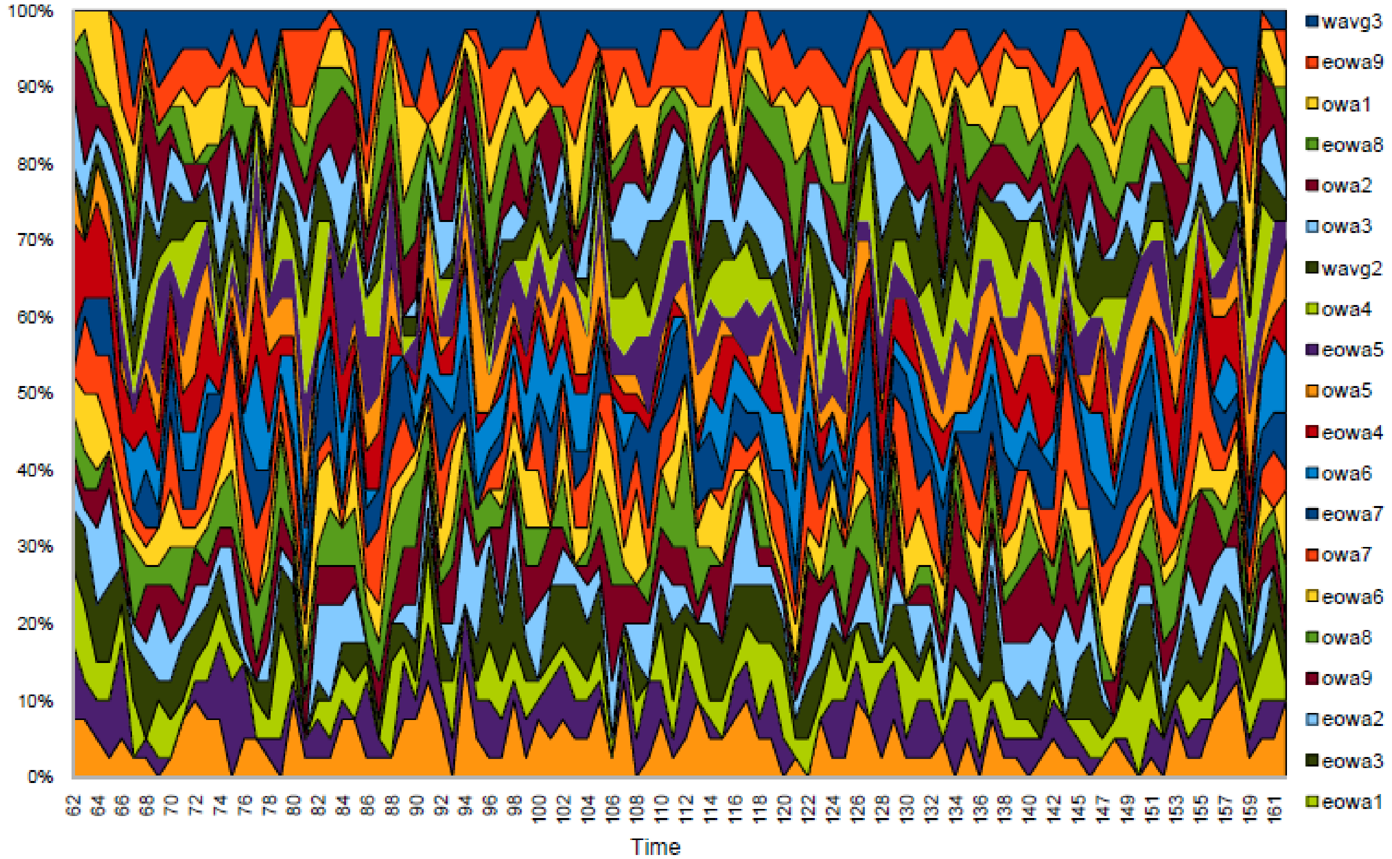
Experimental Evaluation

Dominant strategy behavior, Settings 1



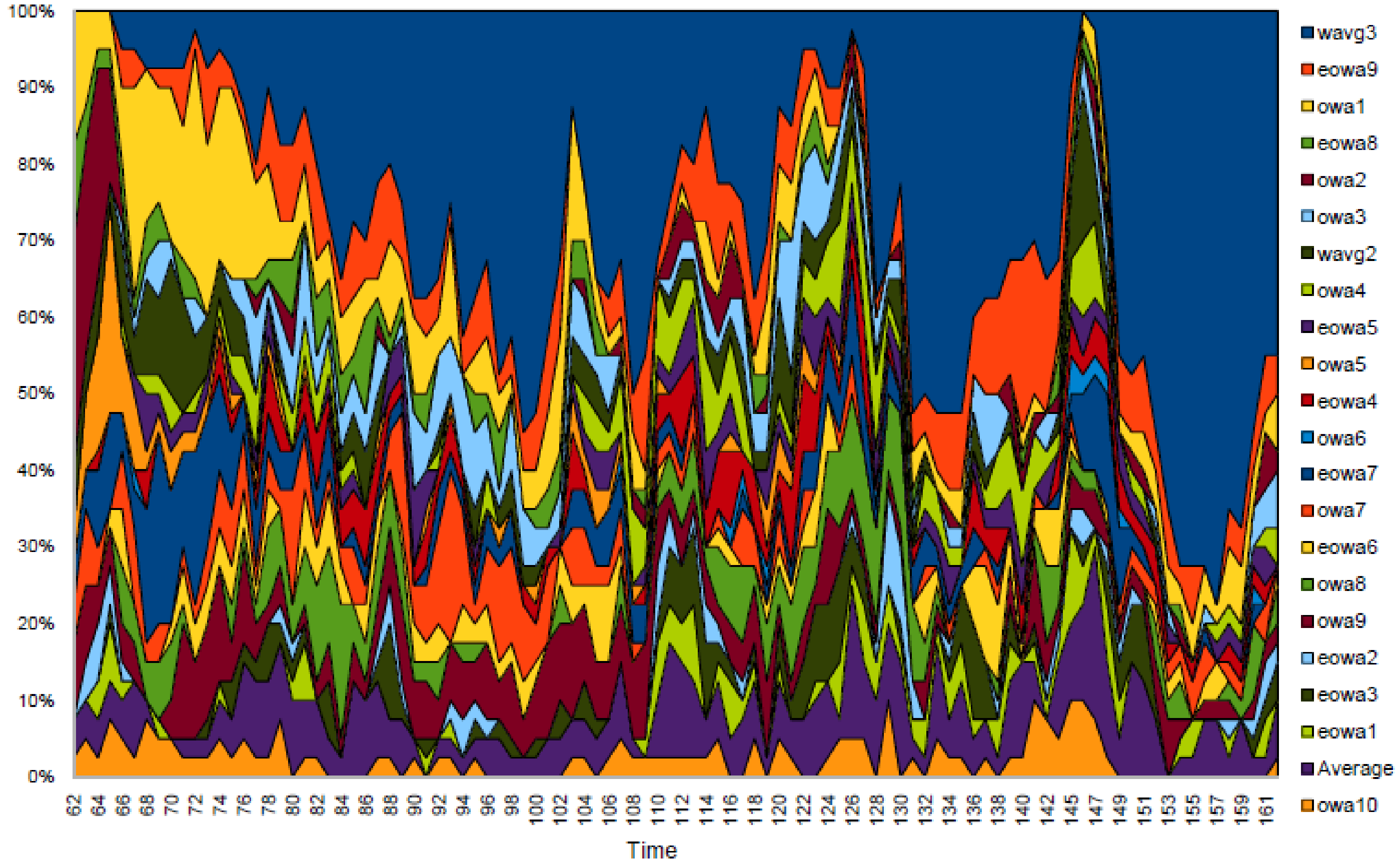
Experimental Evaluation

Conditional dominance, Settings 1



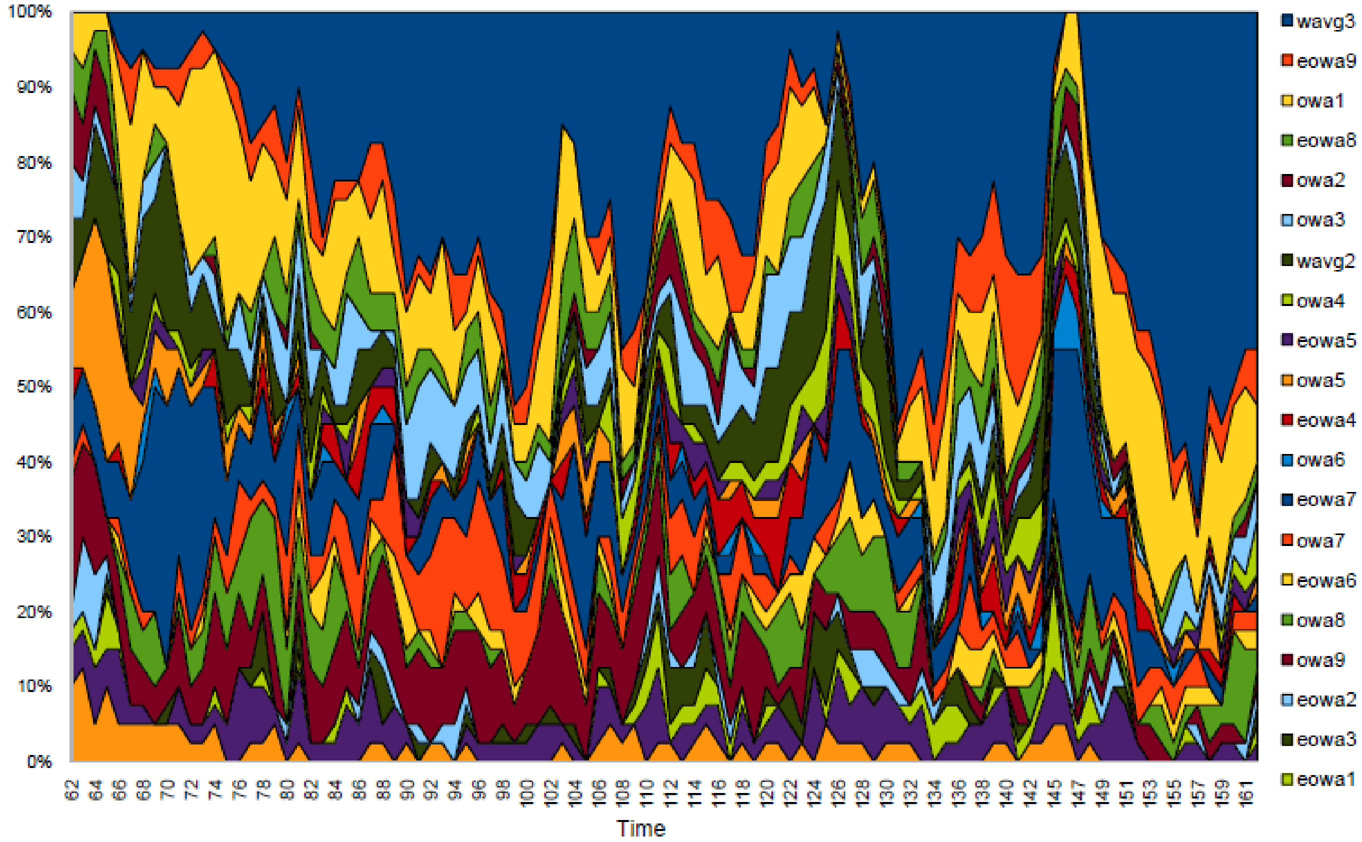
Experimental Evaluation

Max-Min, Settings 1



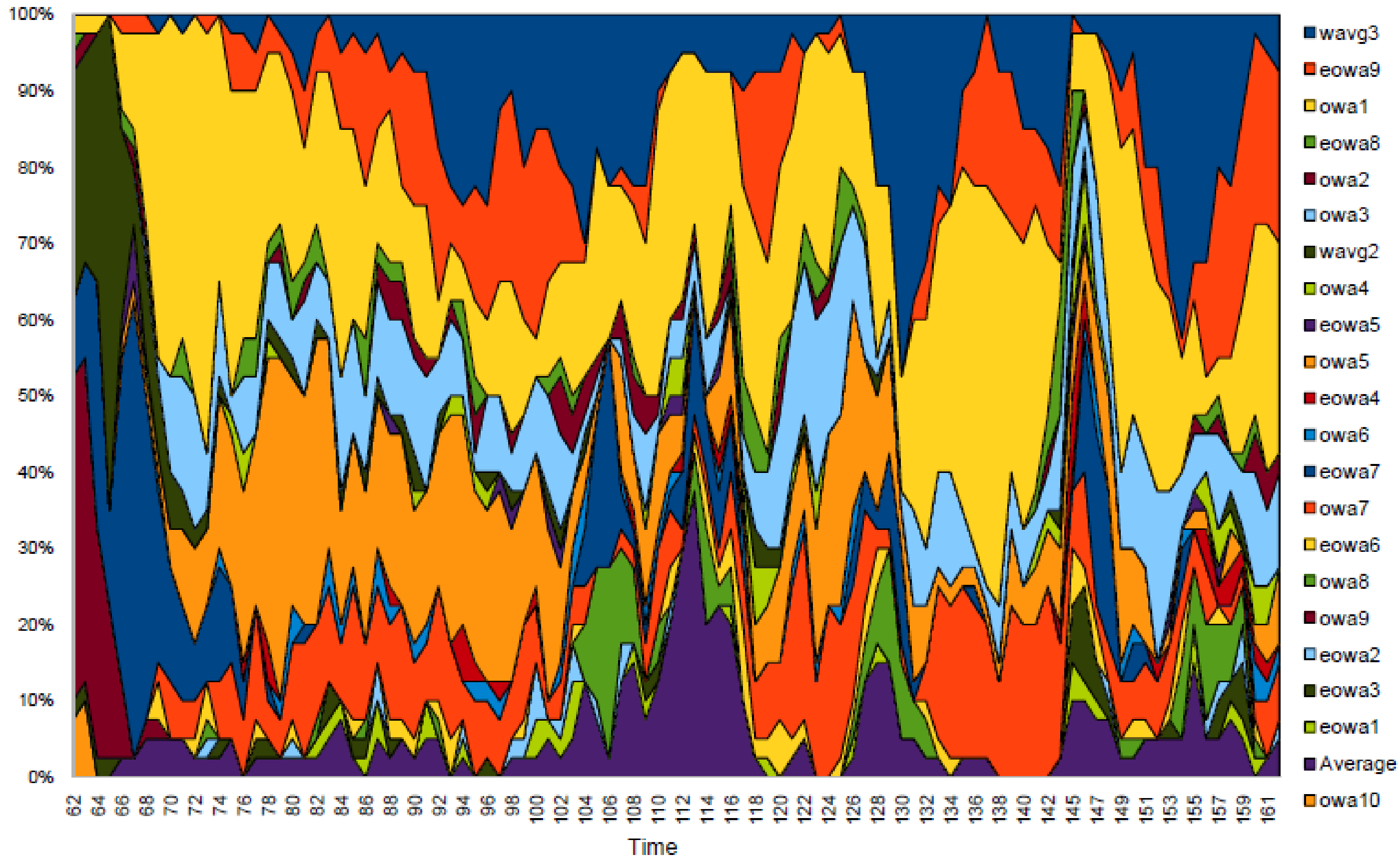
Experimental Evaluation

Nash, Settings 1



Experimental Evaluation

Trust, Settings 1



Experimental Evaluation

Detection payoffs against attack variants; settings 2, False positives included

Attack class	Strategies – settings 2						
	DS	CD	MM	NE	Trust	Best OWA	Avg. OWA
SSH brute force	0.00	13.17	35.01	28.03	18.93	144.68	12.12
Vertical TCP scan with OS detection	2.19	9.66	28.74	27.18	20.68	51.82	6.44
Vertical UDP scan	8.81	12.08	113.32	104.11	59.65	175.61	13.15
Horizontal TCP scan for SSH service	6.69	22.66	85.68	79.99	46.03	130.70	14.92
Horizontal UDP scan for DNS service	14.88	19.02	100.15	91.61	59.75	148.54	14.80
Horizontal ICMP ping scan	23.65	13.99	130.24	135.62	79.20	185.19	11.96
Horizontal TCP scan for all services	0.69	14.94	83.71	69.33	43.44	127.69	16.78
Average	**8.13**	**15.07**	**82.41**	**76.55**	**46.81**	**137.75**	**12.88**

Adaptation Process Attacks

Random selection of challenges

Security policy

Threat models

Network traffic

Algorithms (NBA/trust/adaptation)

Algorithm state/model

Experiments confirmed that:

- Attacker is not much better off with IDS read-only compromise/model
- RNG/challenge sequence needs to be kept secret
- Robustness and *statistical* security is possible

Conclusions

CAMNEP uses **reflective-cognitive** methods to:

- Automatically reduce and maintain the **error rate**
- Monitor system **performance**
- **Optimize** system performance by:
  - Aggregation function selection
  - Aggregation function generation
  - Challenge insertion process management
- Resist **strategic behavior** of informed opponent
  - Game theoretical formalization of the system