A Game-Theoretic Approach for Minimizing Security Risks in the Internet-of-Things

George Rontidis[1], Emmanouil Panaousis[2], Aron Laszka[3], Tasos Dagiuklas[1], Pasquale Malacaria[4], Tansu Alpcan[5]

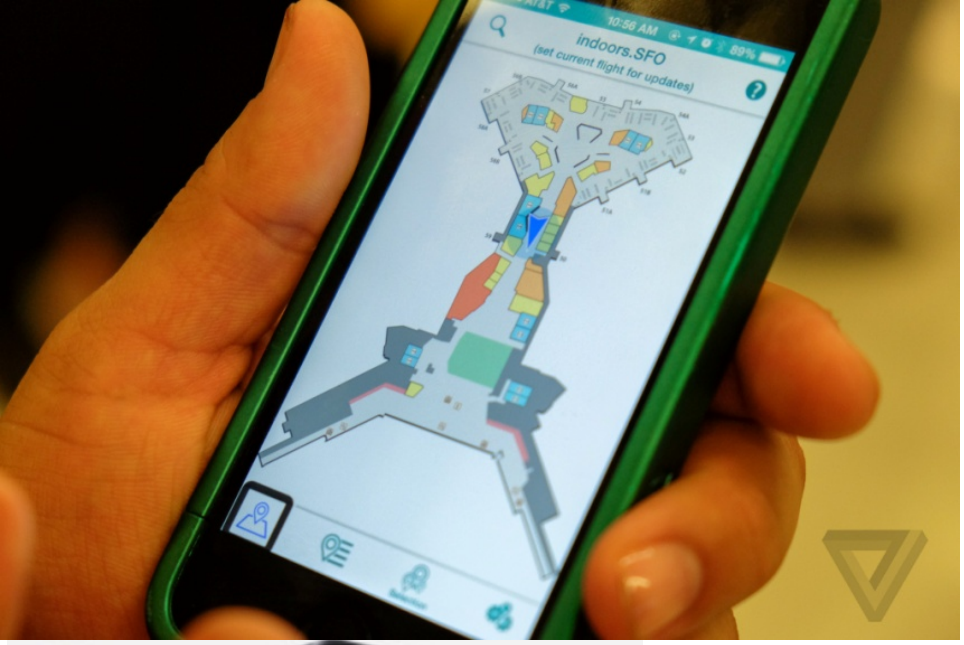
HELLENIC OPEN UNIVERSITY

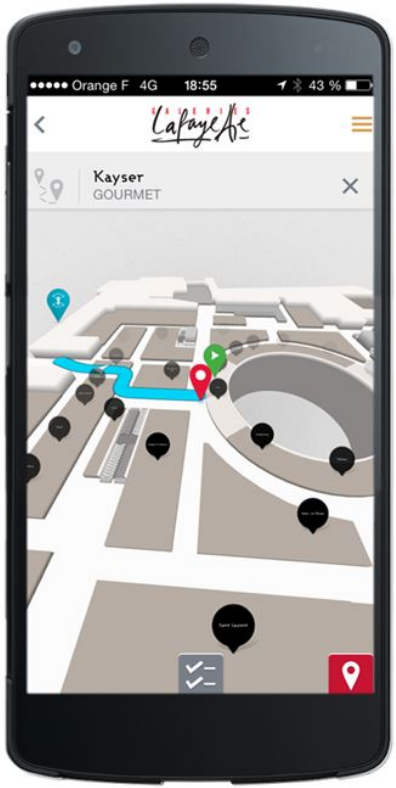
University of Brighton

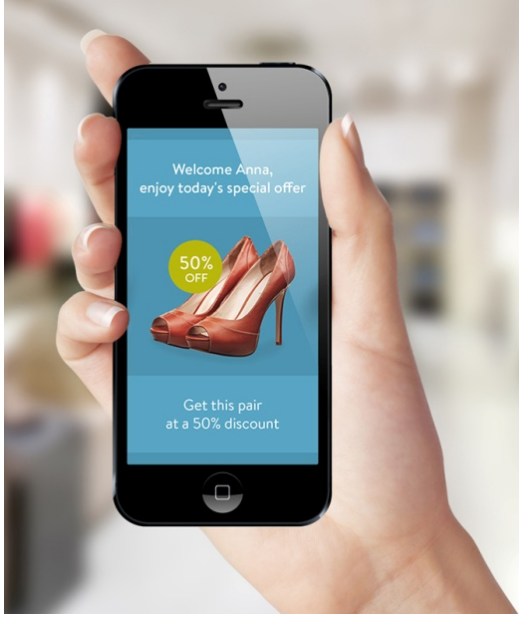
VANDERBILT UNIVERSITY
INSTITUTE FOR SOFTWARE INTEGRATED SYSTEMS

QUEEN MARY UNIVERSITY OF LONDON

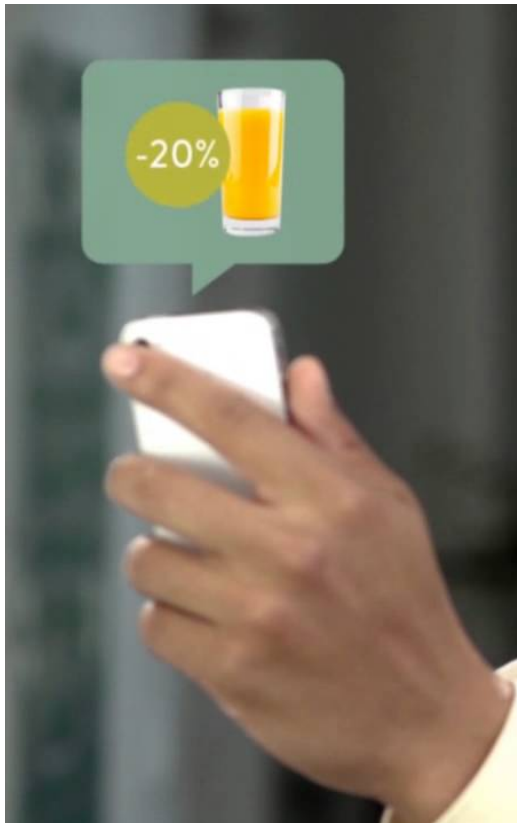
THE UNIVERSITY OF MELBOURNE


indoors.SFO
(set current flight for updates)


Orange F 4G 18:55 43 %
LafayetteAs
Kayser
GOURMET


Welcome Anna,
enjoy today's special offer
50%
OFF
Get this pair
at a 50% discount


Home
Tour
Quiz
Coupons
Settings
Logout
TULPENLAND
FLUWEL


-20%

Motivation (1/2)

- Apps in the IoT enhance user experience
- IoT Prosumers provide services & apps
- “Open environment” → many threats

Motivation (2/2)

- Each IoT prosumer:
  - different services
  - different *sec_level*
- Each user:
  - Connects with 1 or more IoT prosumers (gets services)
  - Shares private data
  - Knows about *sec_level*

[uTRUSTit] Trust Feedback Toolkit

The Problem

- Attacker:
  - Appears as a normal user
  - Can guess the set of prosumers the users prefer
  - Assumes a common sense approach
  - Attacks only one prosumer
- User:
  - Knows about the *sec_level* of each prosumer
  - Selects a subset of prosumers
- Common Sense Approach (CSS):
  - Choose (*k out of N)* Prosumers with higher *sec_level*

Main Contribution

- Decision-support mechanism in selecting a set of prosumers that minimizes security risks in presence of an adversary

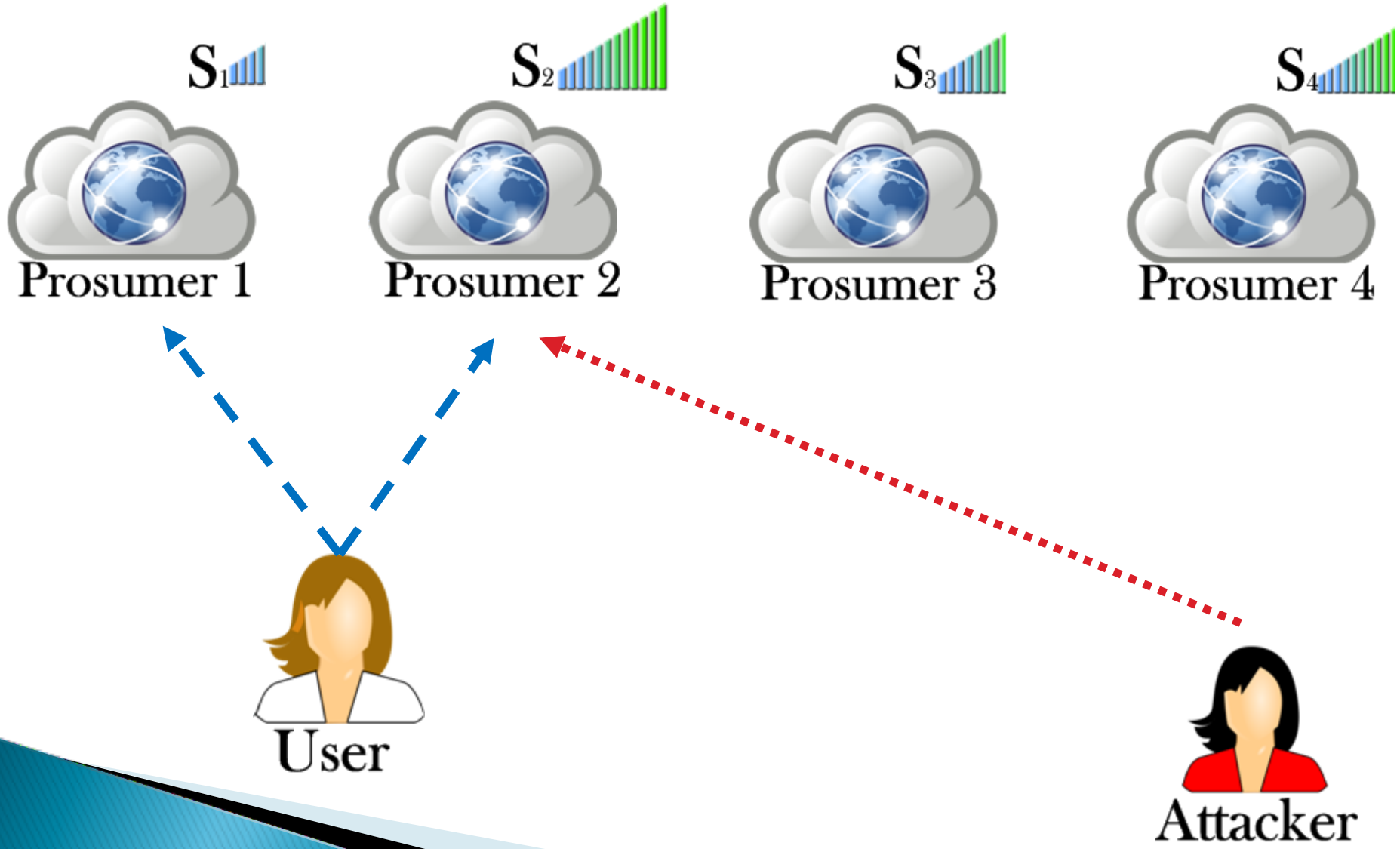
Game Formulation

- Prosumers Selection Game (PSG)
- Two players (*User*, *Attacker*) [abstraction]
- Complete information, zero-sum

The rationale behind the zero-sum game is that when there are clear winners (the attacker) and losers (the defender), and the defender is uncertain about the attacker type, he considers the worst case scenario, which can be formulated by a zero-sum game where the attacker can cause maximum damage to the defender.

- Play simultaneously, [not necessarily during the same timeslot, but when they decide they do not know each other's choice]

Model Illustration



Strategy Sets & Payoffs (1/2)

- Set of Prosumers, P $P := \{1, 2, \ldots, n\}$
- *Sec_level* of Prosumer i, S_i $S_i \quad [0,1)$
- User chooses size-k subset, P' $P' \quad P$
- Attacker attacks i :

Use **latexit** to create nice outcomes for the maths if possible -you can migrate code from the paper

	User	Attacker
$i \in P'$	$-(1-S_i)\,V$	$+(1-S_i)\,V$
$i \notin P'$	no security loss	no benefit

} zero sum

V = user data value

Strategy Sets & Payoffs (2/2)

- Security Risk, R_i $\quad R_i := (1 \quad S_i) \; V$

$(1-S_i)$ → we assume that a more secure prosumer is more difficult to be compromised

Probability of a subset of prosumers to be chosen

Probability of a prosumer to be attacked

	User	Attacker
Pure Strategy	$s = \langle s_i \rangle \quad \{0,1\}^n$	i
Mixed Strategy	$\mathrm{U} = \langle u_s \rangle$	$\mathrm{A} = \langle a_i \rangle$
Expected Payoff	$J_U(\mathrm{s},\mathrm{A}) := \underset{i}{} s_i \, a_i \quad R_i$	$J_A(\mathrm{U},\mathrm{i}) := \underset{s}{} s_i \, u_s \quad R_i$

User's strategic choice influences the payoffs

Usually in GT we write the expected payoffs for a mixed strategy profile – (s,A) and (U,i) are not mixed strategy profiles – just for Tasos not to get confused – no need to revise – next page presents the payoffs for a mixed strategy profile

Theoretical Results

- Given the pair $\langle U, A \rangle$ of mixed strategies:

	User	Attacker
Pure Strategy	$s = \langle s_i \rangle \quad \{0,1\}^n$	i
Mixed Strategy	$\mathrm{U} = \langle u_s \rangle$	$\mathrm{A} = \langle a_i \rangle$
Expected Payoff	$J_U(\mathrm{s},\mathrm{A}) := \sum_i s_i \; a_i \; R_i$	$J_A(\mathrm{U},\mathrm{i}) := \sum_s s_i \; u_s \; R_i$

$$J_U(U,A) = \sum_s \sum_i u_s \; a_i \; s_i \; R_i$$

- If ***p*** is User's strategy, for the pair $\langle p, A \rangle$ of mixed strategies:

$$J_U(p,A) = \sum_i p_i \; a_i \; R_i$$

Nash Equilibrium (NE)

- Two-person game
- Zero-sum game

} at least one NE in mixed strategies

- Finite number of actions
- Saddle point in mixed strategies **(U*,A*)**:

$$U^* = \arg\max_U \min_A J_U(U, A)$$

$$A^* = \arg\max_A \min_U J_A(U, A)$$

- Fundamental game-theoretic result
- Pair of **(U*,A*)** are also called security strategies for players

Nash Defender

- User's strategy at NE: Nash Prosumer Selection (**NPS**)
- Minimax Theorem (1928 John von Neumann):

$$U^* = \arg\min_{U} \max_{A} J_A(U, A)$$

- Regardless of the Attacker's strategy, **NPS** guarantees:
  - A minimum performance
  - An upper limit of expected damage for the User in the presence of a "rational" Attacker (i.e. Attacker who want to play optimally for himself)

- Attacker's strategy: maximize his/her payoff when the User plays $\mathbf{p}$:

$$J_A(\mathbf{p}, i) = J_A(p_i, i) := p_i R_i$$

- Therefore, the support of the Attacker's mixed strategy at the NE has pure strategies (i.e. prosumers to be attacked) that satisfy

$$\arg\max_i \left(p_i R_i \right)$$

NPS Strategy

- Lemma 1: In PSG, for every prosumer i:
  - $p_i=1$ (a prosumer is attacked with absolute certainty) or
  - $p_i R_i=\max_j p_j R_j$ must hold when the User plays the NPS strategy (a prosumer is attacked with probability that is proportional to the maximum payoff of the attacker and inversely proportional to the risk associated with this prosumer [have we defined risk before?] note the latter result makes the difference, because if we did not use game theory we would expect the opposite, i.e. when risk is high the probability of attacking the prosumer is higher. This is because the NE dictates that prosumers with higher risk will be selected more rarely by the User. This is reflected on the following corollary
- Corollary 1: For any NPS strategy and prosumers $i, j \rightarrow R_i \leq R_j \Rightarrow p_i \geq p_j$
- Theorem 1: In PSG, if $k>0$, the User selects every prosumer with some non-zero probability according to NPS (even the least secure one) [this is to spread the risk across the different prosumers and therefore maximizing his expected payoff]

NPS Strategy output

- Assuming that $R_1 \le R_2 \le \ldots \le R_n$ and $k<n$
- Complexity $O(n^2)$

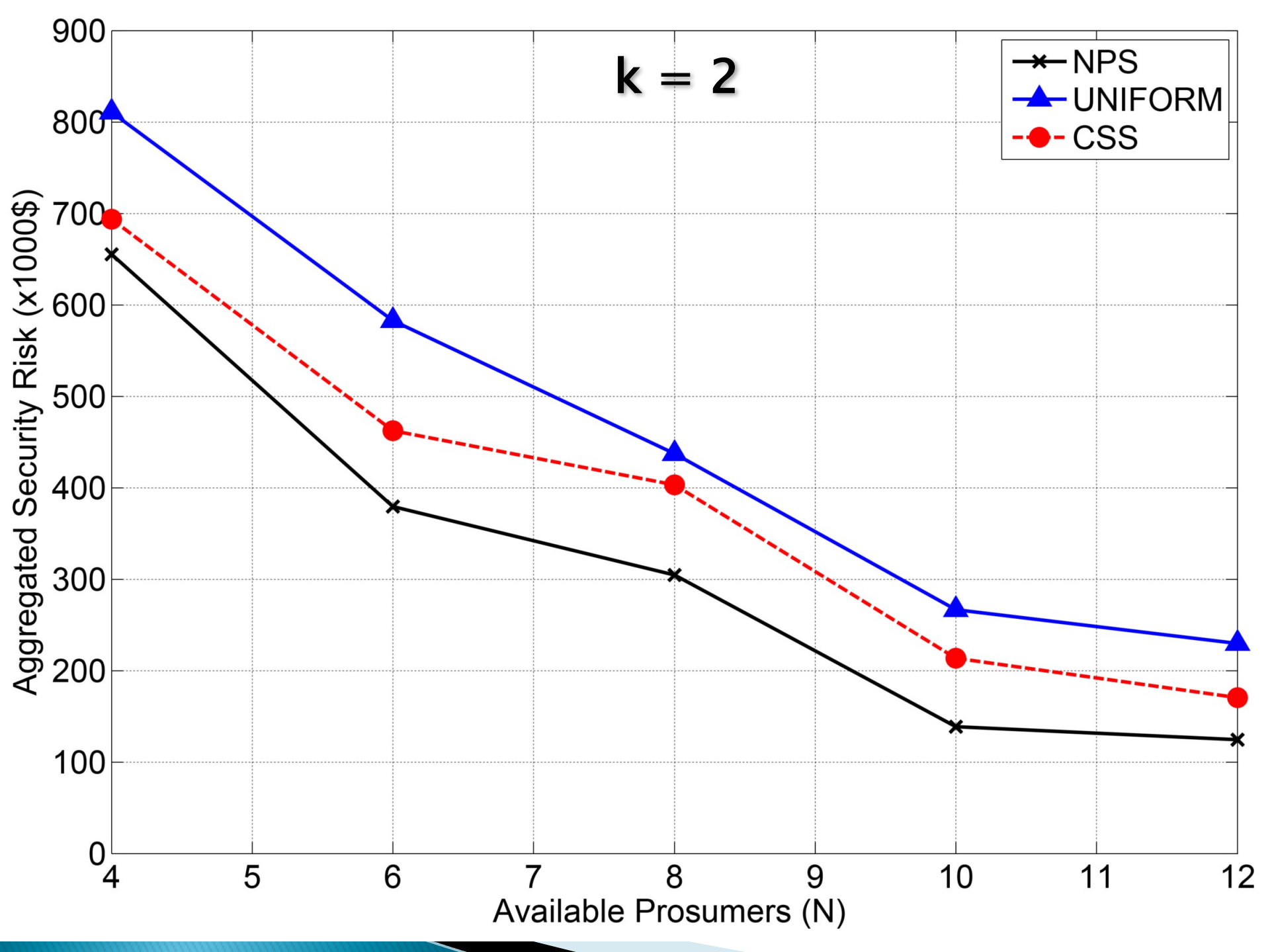
1. *Let S:=k.*
2. *Construct* $p(S) = \langle p_1(S), \ldots, p_n(S) \rangle$ *such that:*
   a) *For every* $i \le S$, *let* $p_i(S) := 1$
   b) *For every* $i > S$, *let* $p_i(S) := (k - S) \dfrac{\frac{1}{R_i}}{\sum_{j=S+1}^{n} \frac{1}{R_j}}$
3. *If* $S=0$ *or* $R_S \le p_{S+1}(S) R_{S+1}$, *then output* $p(S)$
4. *Otherwise, let* $S:=S-1$ *and continue from Step 2.*

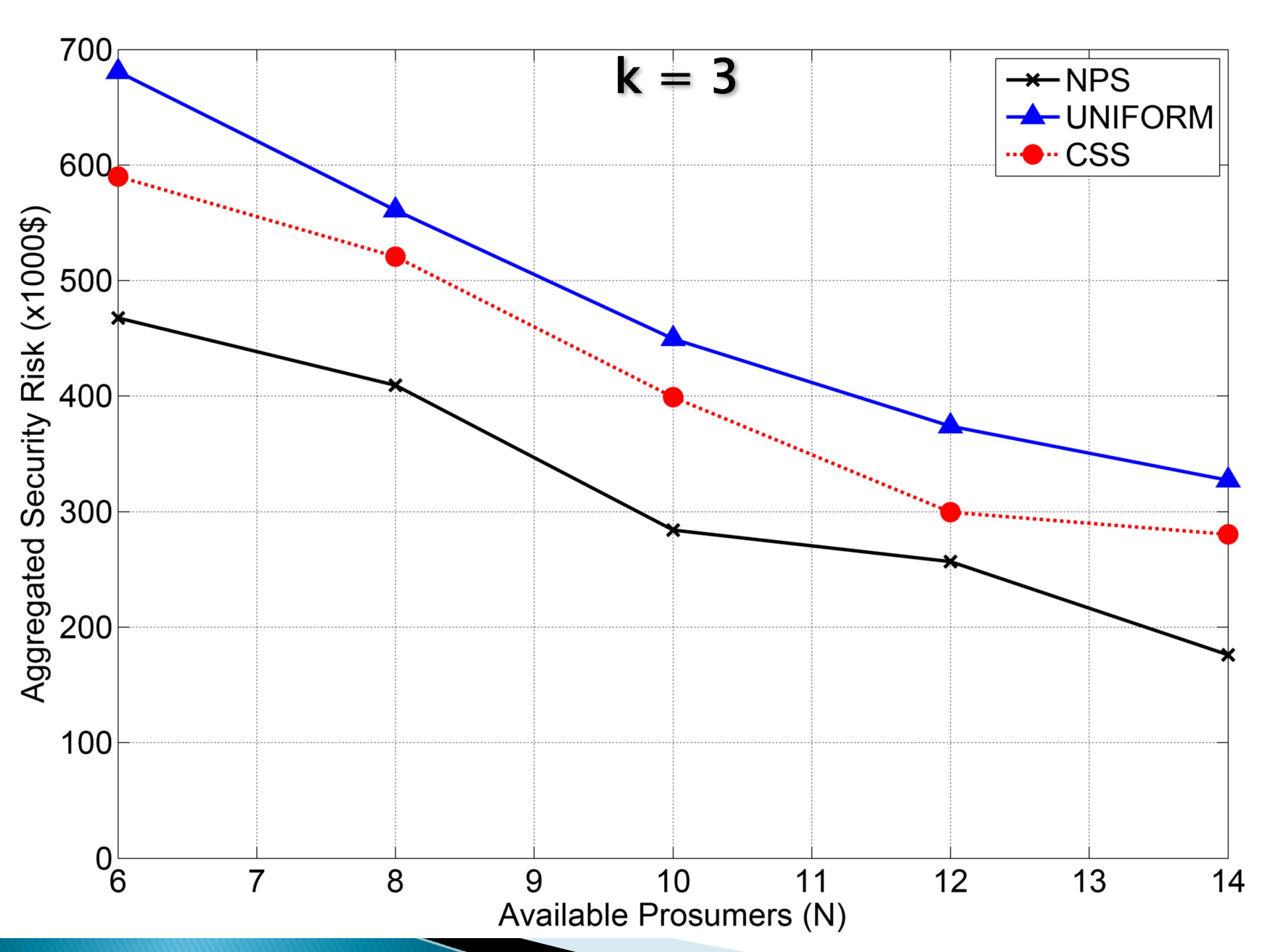
Simulation Results (1/2)

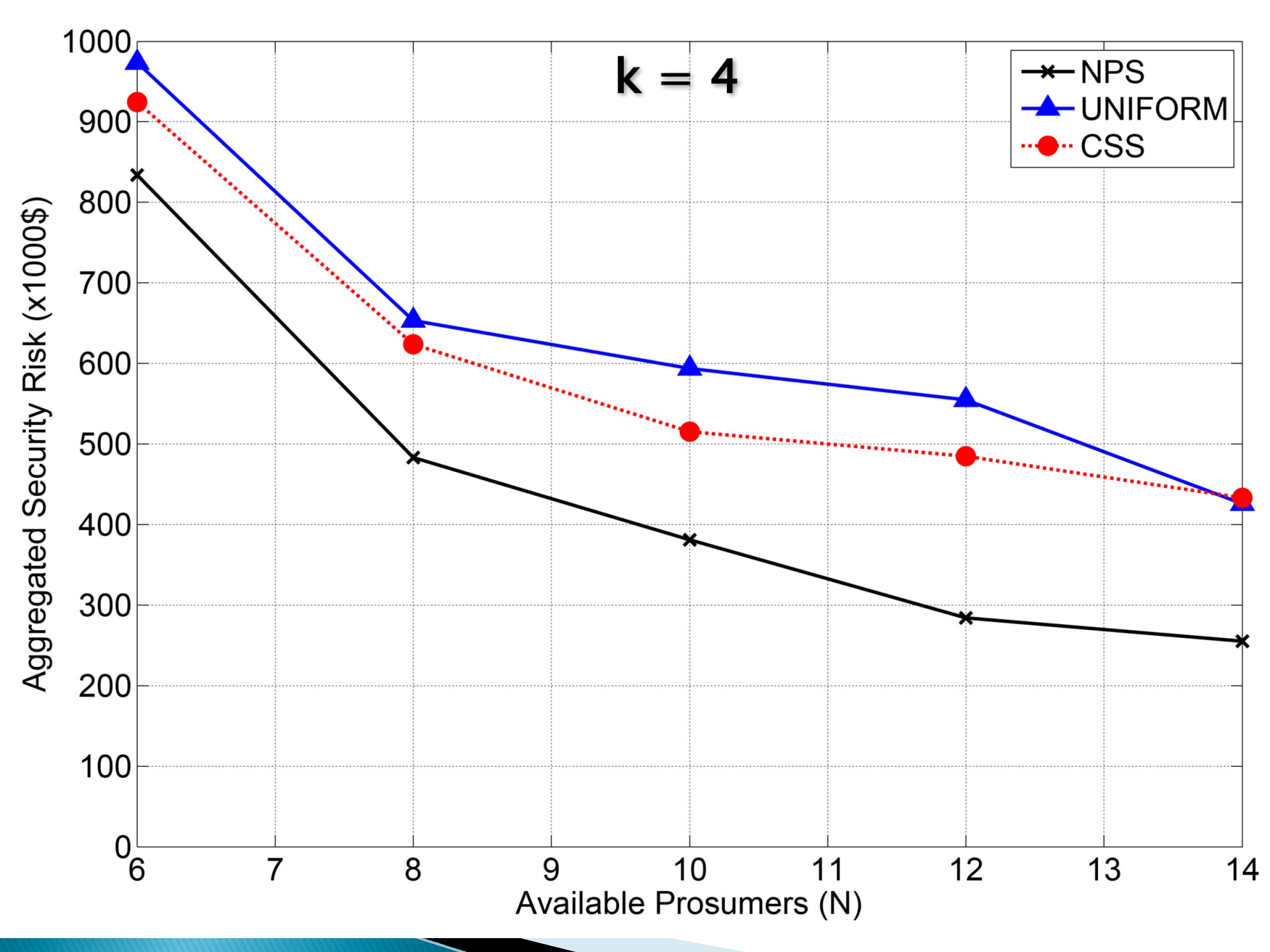
- Comparison of NPS against:
  - Uniform Strategy: selection by using a uniform probability distribution [make clear that uniform refers to the defender]
  - Common Sense Strategy: selection of subset with the most secure prosumers [make clear that uniform refers to the defender – also what does this mean? What is the most secure prosumer – remind to the audience]

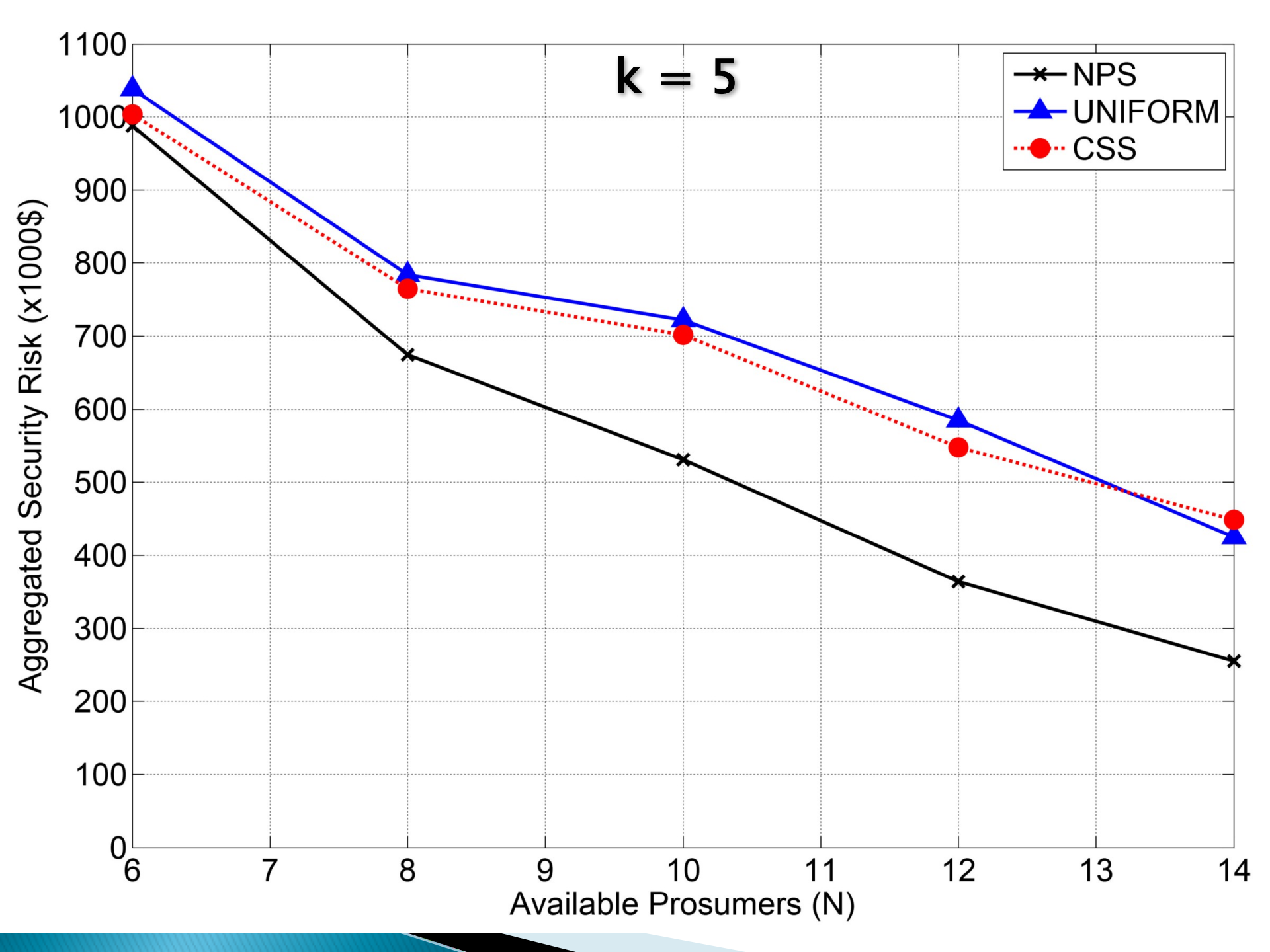
- Rational attacker:
  - Higher sec_level → more likely to be attacked [why ? I would expect the opposite – less secure more likely to be attacked – so basically justify our choice – we must also say that NPS is better regardless of the attacker type because NE=max_{defender strategy} min_{attacker strategy} U_{defender} = min_{defender strategy} max_{attacker strategy} U_{attacker} – this comes from Von Neuman's minimax theorem – Yiorgo make sure Tasos understands this clearly because they will ask him]

- Fixed number of requested prosumers *(k)*
- Variety of available prosumers *(N)*
- 500 selection decisions ~ 500 different users against attacker [they will wonder how different users are captured by one player?]


k = 2
NPS
UNIFORM
CSS
Aggregated Security Risk (x1000$)
Available Prosumers (N)
900
800
700
600
500
400
300
200
100
0
4
5
6
7
8
9
10
11
12


k = 3
NPS
UNIFORM
CSS
Aggregated Security Risk (x1000$)
Available Prosumers (N)
0
100
200
300
400
500
600
700
6
7
8
9
10
11
12
13
14


k = 4
NPS
UNIFORM
CSS
Aggregated Security Risk (x1000$)
Available Prosumers (N)
0
100
200
300
400
500
600
700
800
900
1000
6
7
8
9
10
11
12
13
14


k = 5
NPS
UNIFORM
CSS
Aggregated Security Risk (x1000$)
Available Prosumers (N)
0
100
200
300
400
500
600
700
800
900
1000
1100
6
7
8
9
10
11
12
13
14

Simulation Results (2/2)

- For given k:
  - Security risk ↓ when n ↑
  - [is this not an expected result regardless of game theory? – not interesting]
- For limited user's choices (k→n):
  - Greater security risk is anticipated
  - [is this not an expected result regardless of game theory? – not interesting]
- **NPS always performs better** and achieves on average 1/3 lower security risk

[I cannot also see our algorithm (theorem 2) and theory – basically all lemmas must be presented – these are the core contribution]

Questions