

# KIPDA: $k$ -Indistinguishable Privacy-preserving Data Aggregation in Wireless Sensor Networks

InfoCom 2011

Michael M. Groat\*, Wenbo He<sup>†</sup>, Stephanie Forrest\*

\*Department of Computer Science  
University of New Mexico  
Albuquerque, New Mexico 87131  
United States of America  
Email: {mgroat, forrest}@cs.unm.edu

<sup>†</sup>Department of Electrical Engineering  
University of Nebraska-Lincoln  
Lincoln, Nebraska 68688  
United States of America  
Email: wenbohe@engr.unl.edu

April 13, 2011

# The problem domain

# The problem domain

- Wireless sensor networks:

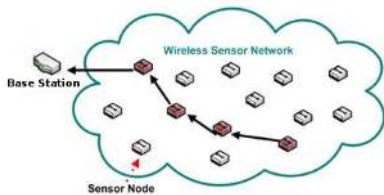


Image from <http://monet.postech.ac.kr/research.html>

# The problem domain

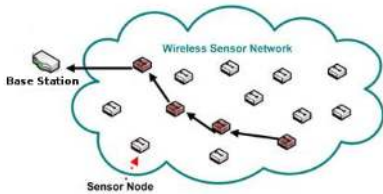


Image from <http://monet.postech.ac.kr/research.html>

- Wireless sensor networks:
  - Network of small resource-constrained devices.

# The problem domain

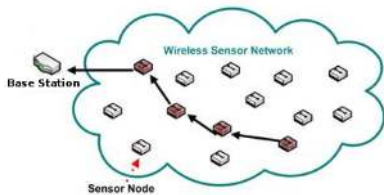


Image from <http://monet.postech.ac.kr/research.html>

- Wireless sensor networks:

- Network of small resource-constrained devices.
- Monitor their environment.

# The problem domain

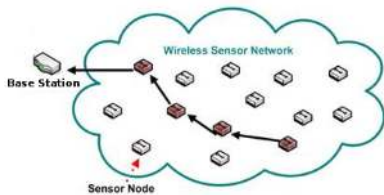


Image from <http://monet.postech.ac.kr/research.html>

- Wireless sensor networks:

- Network of small resource-constrained devices.
- Monitor their environment.
- Limited radio range dictates a hop-by-hop routing topology.

# The problem domain

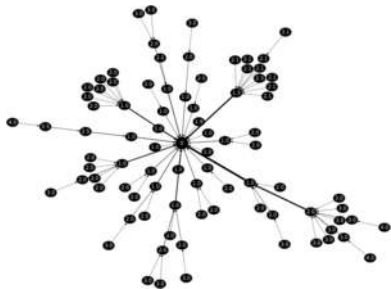


Image from <http://sing.stanford.edu/gnawali/ctp/>

- Wireless sensor networks:
  - Network of small resource-constrained devices.
  - Monitor their environment.
  - Limited radio range dictates a hop-by-hop routing topology.
- Data aggregation:

# The problem domain

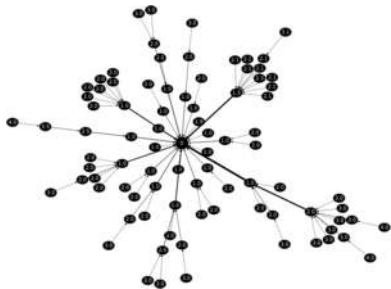


Image from <http://sing.stanford.edu/gnawali/ctp/>

- Wireless sensor networks:
  - Network of small resource-constrained devices.
  - Monitor their environment.
  - Limited radio range dictates a hop-by-hop routing topology.
- Data aggregation:
  - Nodes process, combine, or filter data to conserve bandwidth.



# The problem domain

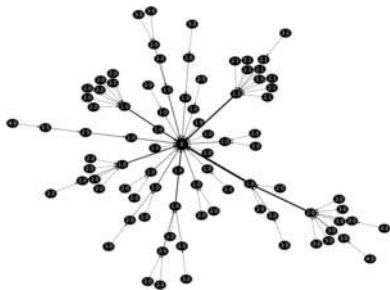


Image from <http://sing.stanford.edu/gnawali/ctp/>

- Wireless sensor networks:
  - Network of small resource-constrained devices.
  - Monitor their environment.
  - Limited radio range dictates a hop-by-hop routing topology.
- Data aggregation:
  - Nodes process, combine, or filter data to conserve bandwidth.
  - We assume a standard tree like routing topology, e.g. the *collection tree protocol*.

# Key challenges with sensitive data

# Key challenges with sensitive data

- Privacy:



Image from

<http://erpfull.com/shop/>

# Key challenges with sensitive data

- Privacy:
  - Data aggregation: more complicated with sensitive data.



Image from

<http://erpfull.com/shop/>

# Key challenges with sensitive data

- Privacy:

- Data aggregation: more complicated with sensitive data.
- We want the nodes to aggregate data.



Image from

<http://erpfull.com/shop/>

# Key challenges with sensitive data

- Privacy:

- Data aggregation: more complicated with sensitive data.
- We want the nodes to aggregate data.
- But we do not want them to know what those data are.



Image from

<http://erpfull.com/shop/>

# Key challenges with sensitive data



Image from

<http://www.freewebs.com/chris343/>

- Privacy:
  - Data aggregation: more complicated with sensitive data.
  - We want the nodes to aggregate data.
  - But we do not want them to know what those data are.
- Power and energy:

# Key challenges with sensitive data



Image from

<http://www.freewebs.com/chris343/>

- Privacy:
  - Data aggregation: more complicated with sensitive data.
  - We want the nodes to aggregate data.
  - But we do not want them to know what those data are.
- Power and energy:
  - Limited amount of power available.



# Key challenges with sensitive data



Image from

<http://www.freewebs.com/chris343/>

- Privacy:
  - Data aggregation: more complicated with sensitive data.
  - We want the nodes to aggregate data.
  - But we do not want them to know what those data are.
- Power and energy:
  - Limited amount of power available.
  - Standard encryption is expensive (computationally, memory, and energy).

# Key challenges with sensitive data



Image from

<http://www.freewebs.com/chris343/>

- Privacy:
  - Data aggregation: more complicated with sensitive data.
  - We want the nodes to aggregate data.
  - But we do not want them to know what those data are.
- Power and energy:
  - Limited amount of power available.
  - Standard encryption is expensive (computationally, memory, and energy).
  - TinySec-AE adds about a 10% increase in energy consumption<sup>1</sup>.

# Key challenges with sensitive data

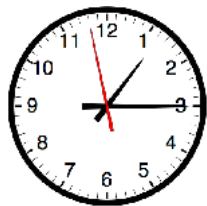


Image from

<http://blog.wolfram.com/2007/07/>

- Privacy:
  - Data aggregation: more complicated with sensitive data.
  - We want the nodes to aggregate data.
  - But we do not want them to know what those data are.
- Power and energy:
  - Limited amount of power available.
  - Standard encryption is expensive (computationally, memory, and energy).
  - TinySec-AE adds about a 10% increase in energy consumption<sup>1</sup>.
- Delay:

# Key challenges with sensitive data

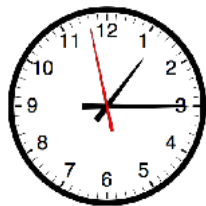


Image from

<http://blog.wolfram.com/2007/07/>

- Privacy:
  - Data aggregation: more complicated with sensitive data.
  - We want the nodes to aggregate data.
  - But we do not want them to know what those data are.
- Power and energy:
  - Limited amount of power available.
  - Standard encryption is expensive (computationally, memory, and energy).
  - TinySec-AE adds about a 10% increase in energy consumption<sup>1</sup>.
- Delay:
  - Nodes need to encrypt a byte in the time to transmit a byte.

# Key challenges with sensitive data

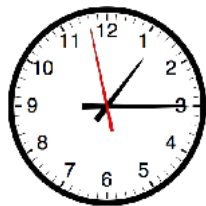


Image from

<http://blog.wolfram.com/2007/07/>

- Privacy:
  - Data aggregation: more complicated with sensitive data.
  - We want the nodes to aggregate data.
  - But we do not want them to know what those data are.
- Power and energy:
  - Limited amount of power available.
  - Standard encryption is expensive (computationally, memory, and energy).
  - TinySec-AE adds about a 10% increase in energy consumption<sup>1</sup>.
- Delay:
  - Nodes need to encrypt a byte in the time to transmit a byte.

<sup>1</sup>C. Karlof, N. Sastry, and D. Wagner. TinySec: A link layer security architecture for wireless sensor networks. *SenSys '04*, 162–175, 2004.

# Addressing these challenges, KIPDA

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.



# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.
- These positions are divided into *restricted* and *unrestricted sets* (and vary between nodes).

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.
- These positions are divided into *restricted* and *unrestricted sets* (and vary between nodes).
- Because aggregates are not encrypted, aggregation can easily take place.

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.
- These positions are divided into *restricted* and *unrestricted sets* (and vary between nodes).
- Because aggregates are not encrypted, aggregation can easily take place.
- Sensitive values are *indistinguishable* from the camouflage values.

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.
- These positions are divided into *restricted* and *unrestricted sets* (and vary between nodes).
- Because aggregates are not encrypted, aggregation can easily take place.
- Sensitive values are *indistinguishable* from the camouflage values.
  - *Definition: An item is indistinguishable from a set of items if an adversary cannot do better than guessing the item from the set.*

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.
- These positions are divided into *restricted* and *unrestricted sets* (and vary between nodes).
- Because aggregates are not encrypted, aggregation can easily take place.
- Sensitive values are *indistinguishable* from the camouflage values.
  - *Definition: An item is indistinguishable from a set of items if an adversary cannot do better than guessing the item from the set.*
- For non-linear functions such as MAX/MIN (can be extended to SUM).

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.
- These positions are divided into *restricted* and *unrestricted sets* (and vary between nodes).
- Because aggregates are not encrypted, aggregation can easily take place.
- Sensitive values are *indistinguishable* from the camouflage values.
  - *Definition: An item is indistinguishable from a set of items if an adversary cannot do better than guessing the item from the set.*
- For non-linear functions such as MAX/MIN (can be extended to SUM).
  - We can not use algebraic properties of polynomials.

# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.
- These positions are divided into *restricted* and *unrestricted sets* (and vary between nodes).
- Because aggregates are not encrypted, aggregation can easily take place.
- Sensitive values are *indistinguishable* from the camouflage values.
  - *Definition: An item is indistinguishable from a set of items if an adversary cannot do better than guessing the item from the set.*
- For non-linear functions such as MAX/MIN (can be extended to SUM).
  - We can not use algebraic properties of polynomials.
  - Homomorphic encryption does not work.



# Addressing these challenges, KIPDA

KIPDA: k-Indistinguishable Privacy-preserving Data Aggregation:

- Aggregates are anonymized among camouflage data in a *message set*.
- The values in certain positions in the message set obey special properties.
- These positions are divided into *restricted* and *unrestricted sets* (and vary between nodes).
- Because aggregates are not encrypted, aggregation can easily take place.
- Sensitive values are *indistinguishable* from the camouflage values.
  - *Definition: An item is indistinguishable from a set of items if an adversary cannot do better than guessing the item from the set.*
- For non-linear functions such as MAX/MIN (can be extended to SUM).
  - We can not use algebraic properties of polynomials.
  - Homomorphic encryption does not work.
  - Perturbation techniques are not applicable.

# KIPDA's privacy assumptions and threat model

---

1

# KIPDA's privacy assumptions and threat model

- Privacy assumptions:

# KIPDA's privacy assumptions and threat model

- Privacy assumptions:
  - A datum is *k-indistinguishable* from  $k - 1$  other camouflage data.

# KIPDA's privacy assumptions and threat model

- Privacy assumptions:
  - A datum is *k-indistinguishable* from  $k - 1$  other camouflage data.
    - Definition: An item is *k-indistinguishable* if it cannot be distinguished better than guessing from  $k - 1$  other items.

# KIPDA's privacy assumptions and threat model

- Privacy assumptions:
  - A datum is *k-indistinguishable* from  $k - 1$  other camouflage data.
    - Definition: An item is *k-indistinguishable* if it cannot be distinguished better than guessing from  $k - 1$  other items.
  - A certain level of node collusion or capture is tolerated.

# KIPDA's privacy assumptions and threat model

- Privacy assumptions:
  - A datum is *k-indistinguishable* from  $k - 1$  other camouflage data.
    - Definition: An item is *k-indistinguishable* if it cannot be distinguished better than guessing from  $k - 1$  other items.
  - A certain level of node collusion or capture is tolerated.
- Threat model includes threats from:

# KIPDA's privacy assumptions and threat model

- Privacy assumptions:
  - A datum is *k-indistinguishable* from  $k - 1$  other camouflage data.
    - Definition: An item is *k-indistinguishable* if it cannot be distinguished better than guessing from  $k - 1$  other items.
  - A certain level of node collusion or capture is tolerated.
- Threat model includes threats from:
  - Untrusted eavesdroppers intercepting or listening to packets.



# KIPDA's privacy assumptions and threat model

- Privacy assumptions:
  - A datum is *k-indistinguishable* from  $k - 1$  other camouflage data.
    - Definition: An item is *k-indistinguishable* if it cannot be distinguished better than guessing from  $k - 1$  other items.
  - A certain level of node collusion or capture is tolerated.
- Threat model includes threats from:
  - Untrusted eavesdroppers intercepting or listening to packets.
  - *Honest but curious*<sup>1</sup> nodes in between data transit.

---

<sup>1</sup> V. Bozovic, D. Socek, R. Steinwandt, and V. I. Villanyi. Multi-authority attribute based encryption with honest-but-curious central authority. *IACR eprint archive*, 2009.

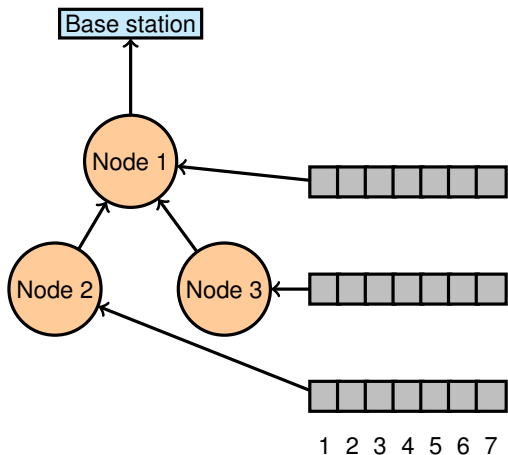
# KIPDA's privacy assumptions and threat model

- Privacy assumptions:
  - A datum is *k-indistinguishable* from  $k - 1$  other camouflage data.
    - Definition: An item is *k-indistinguishable* if it cannot be distinguished better than guessing from  $k - 1$  other items.
  - A certain level of node collusion or capture is tolerated.
- Threat model includes threats from:
  - Untrusted eavesdroppers intercepting or listening to packets.
  - *Honest but curious*<sup>1</sup> nodes in between data transit.
  - Polynomial time adversaries.

---

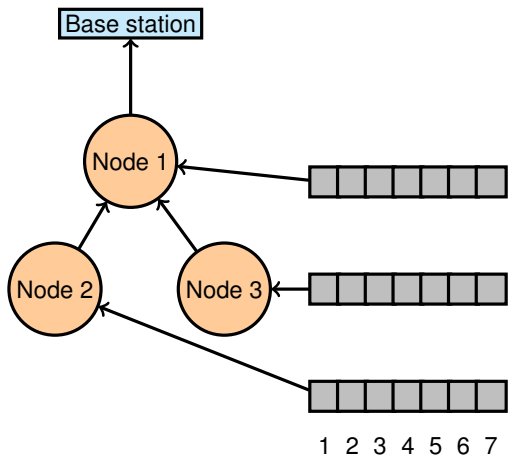
<sup>1</sup> V. Bozovic, D. Socek, R. Steinwandt, and V. I. Villanyi. Multi-authority attribute based encryption with honest-but-curious central authority. *IACR eprint archive*, 2009.

# KIPDA example for MAX aggregation



KIPDA example (MAX aggregation)

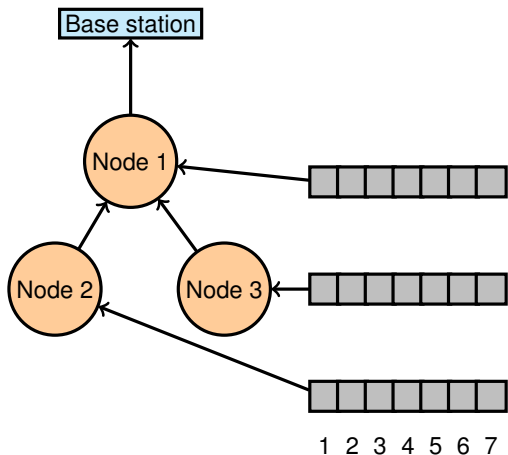
# KIPDA example for MAX aggregation



## KIPDA example (MAX aggregation)

- Nodes 2 and 3 report to node 1, who reports to the base station.

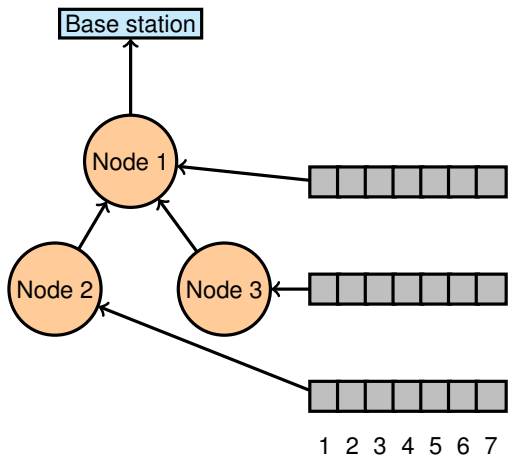
# KIPDA example for MAX aggregation



## KIPDA example (MAX aggregation)

- Nodes 2 and 3 report to node 1, who reports to the base station.
- Each node wants to report one number, keeping that number anonymous.

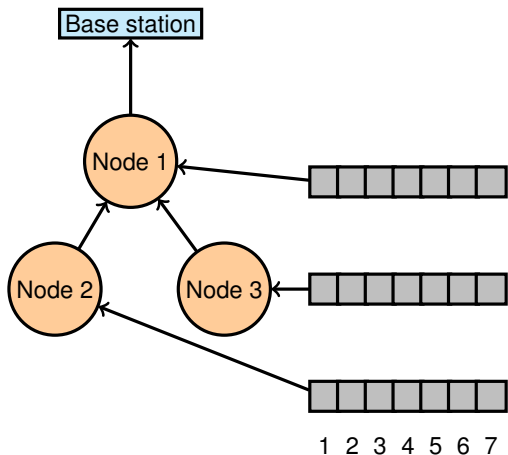
# KIPDA example for MAX aggregation



## KIPDA example (MAX aggregation)

- Nodes 2 and 3 report to node 1, who reports to the base station.
- Each node wants to report one number, keeping that number anonymous.
- KIPDA makes that number indistinguishable from the others.

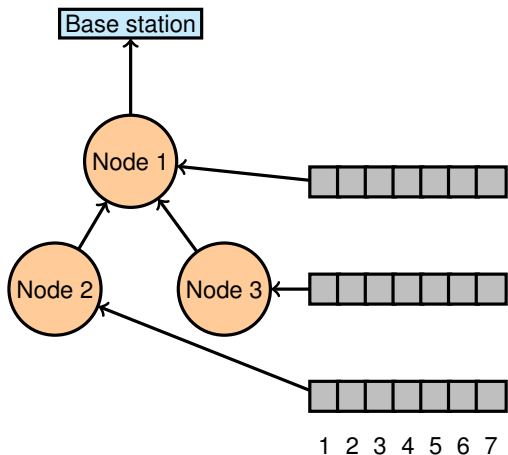
# KIPDA example for MAX aggregation



## KIPDA example (MAX aggregation)

- Nodes 2 and 3 report to node 1, who reports to the base station.
- Each node wants to report one number, keeping that number anonymous.
- KIPDA makes that number indistinguishable from the others.
- Message set of size 7.

# KIPDA example for MAX aggregation

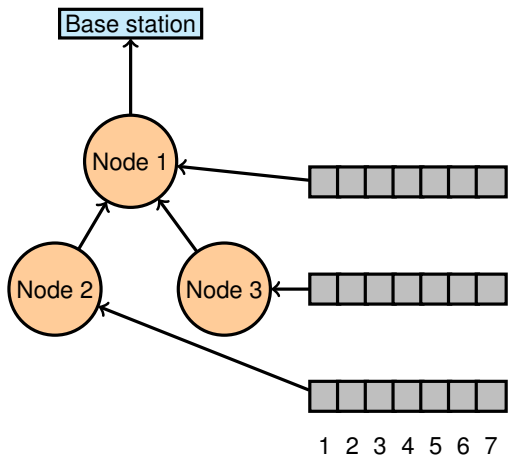


4 phases to the protocol:

- 1 Pre-deployment phase.
- 2 Reporting phase.
- 3 Aggregation phase.
- 4 Base-station processing phase.

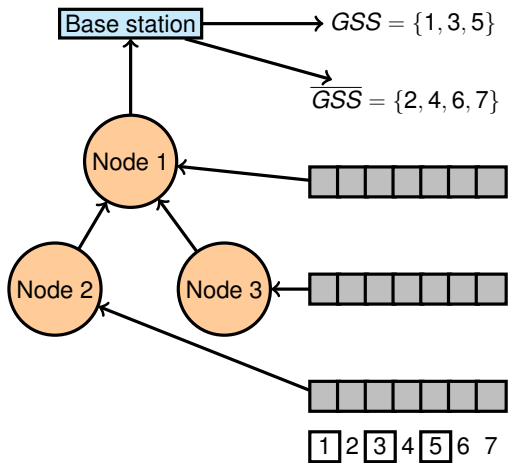


# KIPDA example for MAX aggregation



1) Pre-deployment phase:

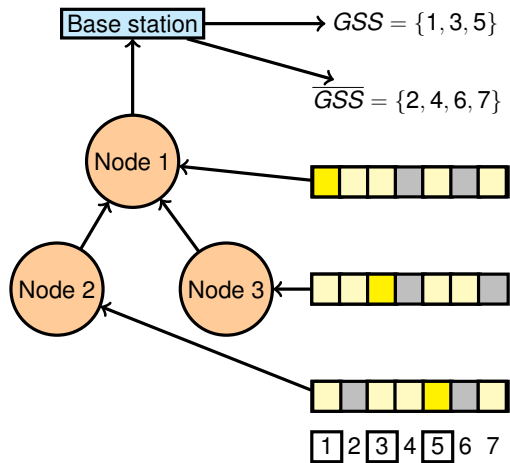
# KIPDA example for MAX aggregation



1) Pre-deployment phase:

- BS chooses the size for the *global secret set*, (*GSS*), then fills it in.

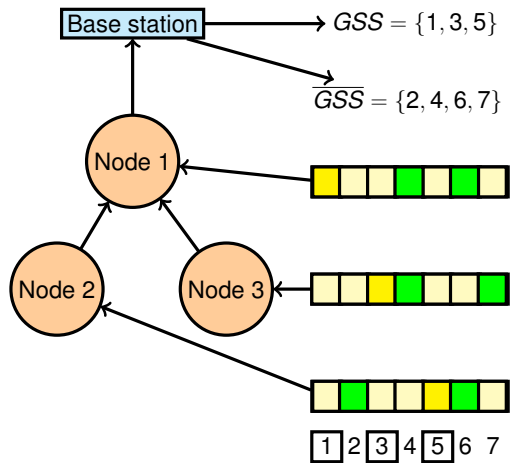
# KIPDA example for MAX aggregation



## 1) Pre-deployment phase:

- BS chooses the size for the *global secret set*, ( $GSS$ ), then fills it in.
- BS distributes the restricted sets, ( $RS_i$ ), to each node  $i$ . (Yellow shades).
  - 1  $GSS \subset RS_i$  (Accuracy).
  - 2  $RS_i \subset \overline{GSS}$  (Anonymity).
  - 3 Truth value position  $\in GSS$  (Accuracy).

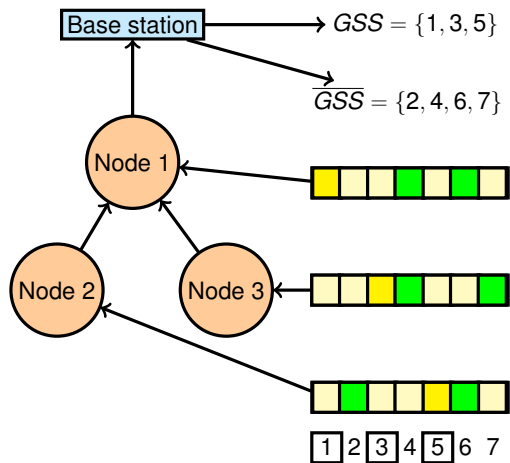
# KIPDA example for MAX aggregation



## 1) Pre-deployment phase:

- BS chooses the size for the *global secret set*, ( $GSS$ ), then fills it in.
- BS distributes the restricted sets, ( $RS_i$ ), to each node  $i$ . (Yellow shades).
  - 1  $GSS \subset RS_i$  (Accuracy).
  - 2  $RS_i \subset \overline{GSS}$  (Anonymity).
  - 3 Truth value position  $\in GSS$  (Accuracy).
- Nodes trivially determine unrestricted sets (Green).

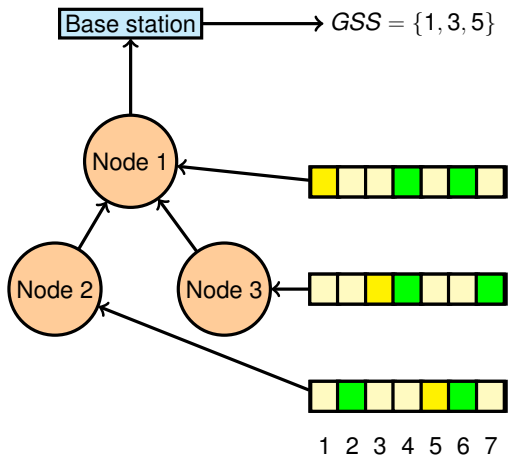
# KIPDA example for MAX aggregation



## 1) Pre-deployment phase:

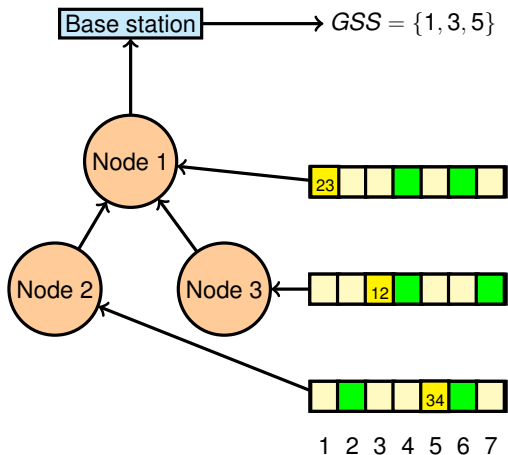
- BS chooses the size for the *global secret set*, ( $GSS$ ), then fills it in.
- BS distributes the restricted sets, ( $RS_i$ ), to each node  $i$ . (Yellow shades).
  - 1  $GSS \subset RS_i$  (Accuracy).
  - 2  $RS_i \subset \overline{GSS}$  (Anonymity).
  - 3 Truth value position  $\in GSS$  (Accuracy).
- Nodes trivially determine unrestricted sets (Green).
- Attention is given to the sizes of sets.

# KIPDA example for MAX aggregation



2) Reporting phase:

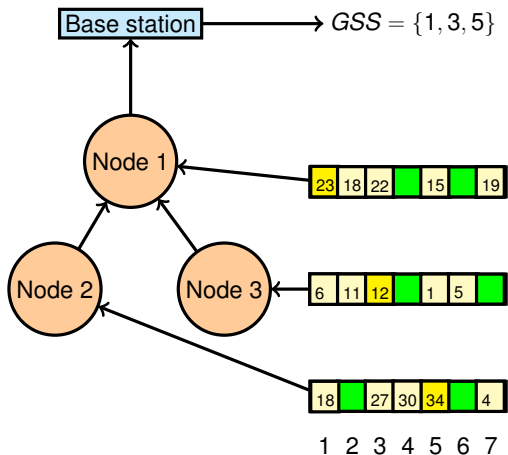
# KIPDA example for MAX aggregation



## 2) Reporting phase:

- The sensed values are put in the real value slots, (dark yellow).

# KIPDA example for MAX aggregation

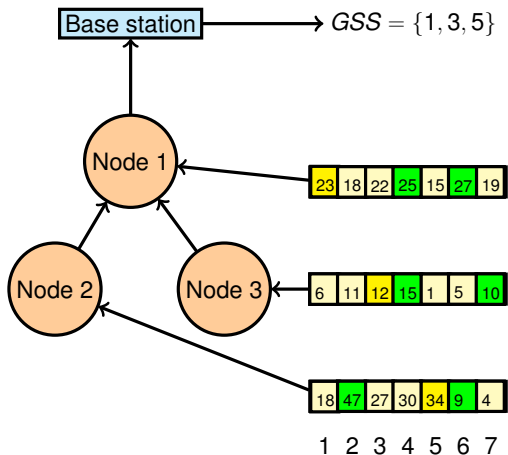


## 2) Reporting phase:

- The sensed values are put in the real value slots, (dark yellow).
- Restricted slots are filled with values that below the sensed value.



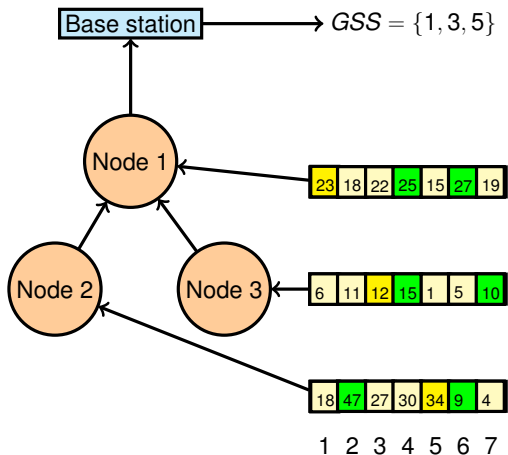
# KIPDA example for MAX aggregation



## 2) Reporting phase:

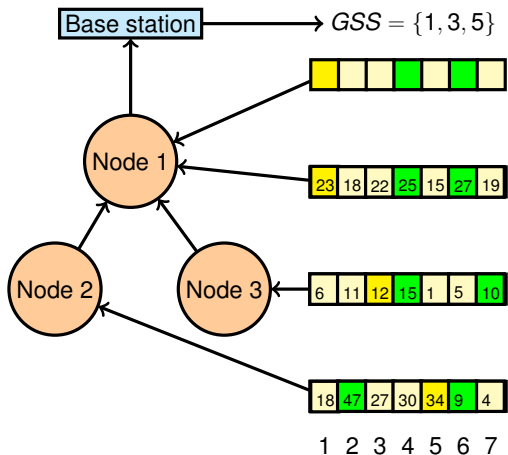
- The sensed values are put in the real value slots, (dark yellow).
- Restricted slots are filled with values that below the sensed value.
- Unrestricted slots are filled with values either above or below the sensed value.

# KIPDA example for MAX aggregation



3) Aggregation phase:

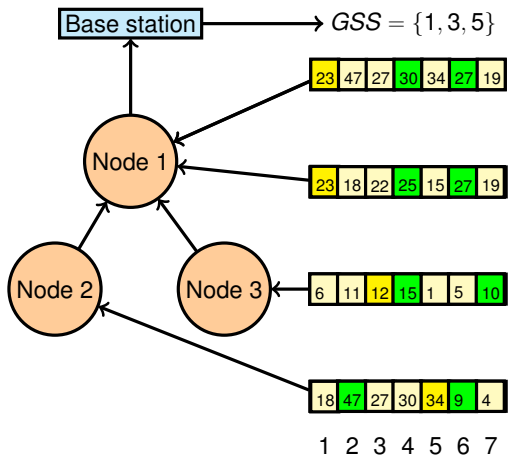
# KIPDA example for MAX aggregation



## 3) Aggregation phase:

- The aggregation function is then performed on the children and itself, if the aggregator senses.

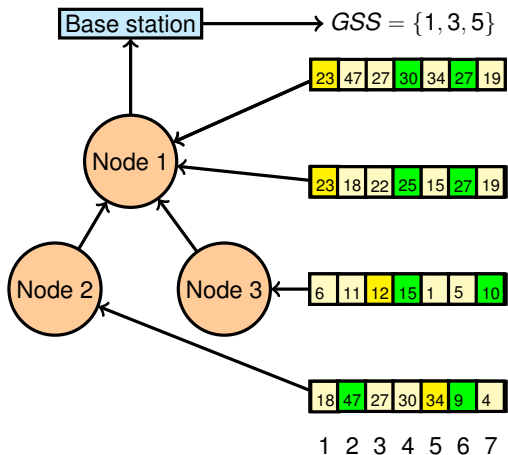
# KIPDA example for MAX aggregation



## 3) Aggregation phase:

- The aggregation function is then performed on the children and itself, if the aggregator senses.
- The MAX is taken from all three message sets for each position.

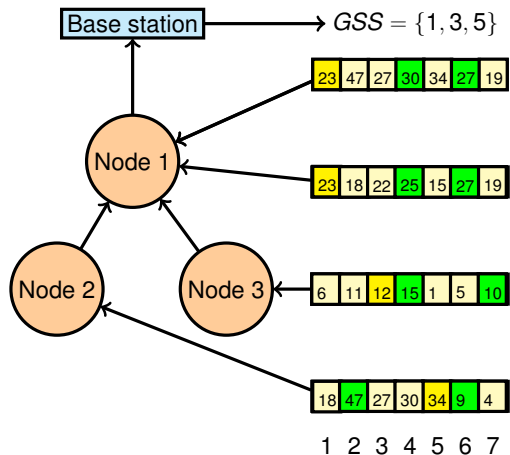
# KIPDA example for MAX aggregation



## 3) Aggregation phase:

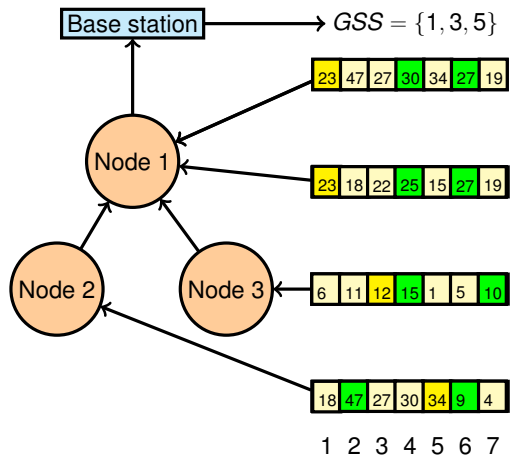
- The aggregation function is then performed on the children and itself, if the aggregator senses.
- The MAX is taken from all three message sets for each position.
- Message set is sent up the aggregation tree.

# KIPDA example for MAX aggregation



4) Base station phase:

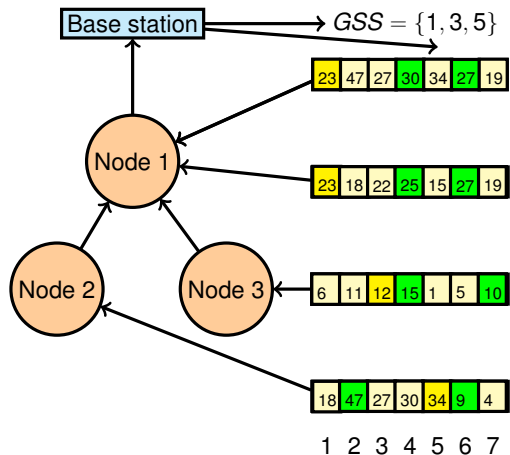
# KIPDA example for MAX aggregation



## 4) Base station phase:

- The base station determines the network aggregate by taking the maximum from the  $GSS$ .

# KIPDA example for MAX aggregation



## 4) Base station phase:

- The base station determines the network aggregate by taking the maximum from the  $GSS$ .
- Position 5 contains the maximum.



# KIPDA: other aggregation functions

- Summation aggregation function:

# KIPDA: other aggregation functions

- Summation aggregation function:
  - Truth values: more than one.

# KIPDA: other aggregation functions

- Summation aggregation function:
  - Truth values: more than one.
  - Truth values: sum to sensed value.

# KIPDA: other aggregation functions

- Summation aggregation function:
  - Truth values: more than one.
  - Truth values: sum to sensed value.
  - Restricted values: sum to 0.

# KIPDA: other aggregation functions

- Summation aggregation function:
  - Truth values: more than one.
  - Truth values: sum to sensed value.
  - Restricted values: sum to 0.
  - Unrestricted values: sum to any value.

# But does this save energy?

# But does this save energy?

- Even though more messages are transmitted, energy is conserved.

# But does this save energy?

- Even though more messages are transmitted, energy is conserved.
- We determined the energy to encrypt and decrypt by IDEA, RC4, and RC5.



# But does this save energy?

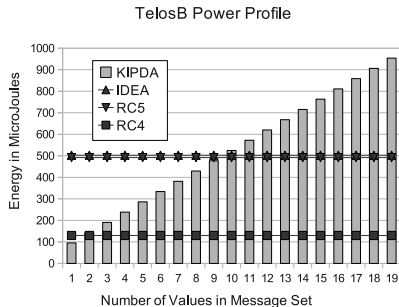
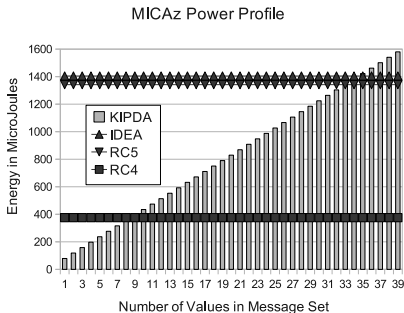
- Even though more messages are transmitted, energy is conserved.
- We determined the energy to encrypt and decrypt by IDEA, RC4, and RC5.
- We then extrapolated this to a standard hop-by-hop encryption scheme.

# But does this save energy?

- Even though more messages are transmitted, energy is conserved.
- We determined the energy to encrypt and decrypt by IDEA, RC4, and RC5.
- We then extrapolated this to a standard hop-by-hop encryption scheme.
- And then applied this to two common architectures, MICAz and TelosB.

# But does this save energy?

- Even though more messages are transmitted, energy is conserved.
- We determined the energy to encrypt and decrypt by IDEA, RC4, and RC5.
- We then extrapolated this to a standard hop-by-hop encryption scheme.
- And then applied this to two common architectures, MICAz and TelosB.



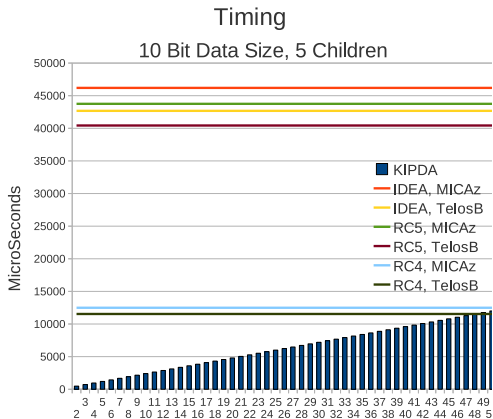
# And this also saves time!

# And this also saves time!

- KIPDA excels in timing, saving on the network delay:

# And this also saves time!

- KIPDA excels in timing, saving on the network delay:



# Privacy guarantees

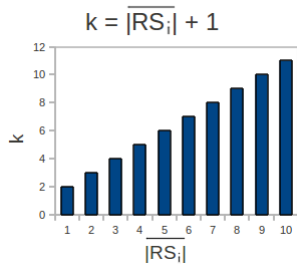
# Privacy guarantees

- Privacy is quantified by the level of  $k$ .



# Privacy guarantees

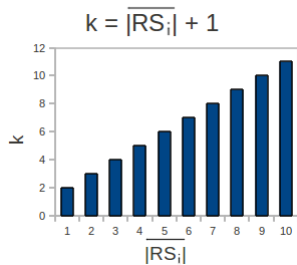
- Privacy is quantified by the level of  $k$ .
- $k$  is given as:



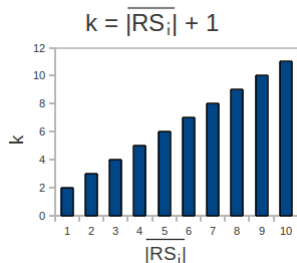
# Privacy guarantees

- Privacy is quantified by the level of  $k$ .
- $k$  is given as:

$$k = |\overline{RS}_i| + 1.$$



# Privacy guarantees

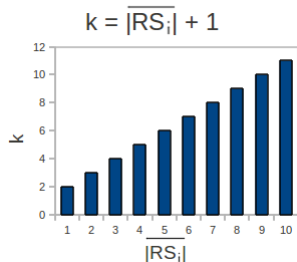


- Privacy is quantified by the level of  $k$ .
- $k$  is given as:

$$k = |\overline{RS}_i| + 1.$$

- Any node  $i$  knows for any node  $j$  the real value is in the  $|\overline{RS}_i| + 1$  largest values.

# Privacy guarantees

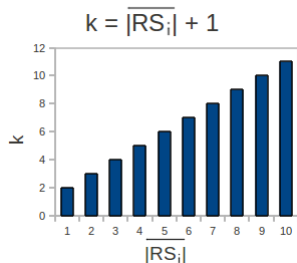


- Privacy is quantified by the level of  $k$ .
- $k$  is given as:

$$k = |\overline{RS}_i| + 1.$$

- Any node  $i$  knows for any node  $j$  the real value is in the  $|\overline{RS}_i| + 1$  largest values.
- To an outside observer though,  $k$  equals the size of the message set.

# Privacy guarantees



- Privacy is quantified by the level of  $k$ .
- $k$  is given as:

$$k = |\overline{RS}_i| + 1.$$

- Any node  $i$  knows for any node  $j$  the real value is in the  $|\overline{RS}_i| + 1$  largest values.
- To an outside observer though,  $k$  equals the size of the message set.
- $k$  is reduced if more rogue nodes collude.

# Privacy: Encryption vs. KIPDA

# Privacy: Encryption vs. KIPDA

Method

Limitations

---

# Privacy: Encryption vs. KIPDA

Method

Limitations

---

Hob-by-hop Encryption    1) Aggregate data are vulnerable at the nodes.



# Privacy: Encryption vs. KIPDA

Method	Limitations
Hop-by-hop Encryption	<ol style="list-style-type: none"><li data-bbox="622 298 1648 350">1) Aggregate data are vulnerable at the nodes.</li><li data-bbox="622 360 1758 409">2) Does not work well for honest-but-curious nodes.</li></ol>

# Privacy: Encryption vs. KIPDA

Method	Limitations
Hob-by-hop Encryption	1) Aggregate data are vulnerable at the nodes. 2) Does not work well for honest-but-curious nodes.
End-to-End Encryption	1) Does not work well for non-linear functions.

# Privacy: Encryption vs. KIPDA

Method	Limitations
Hob-by-hop Encryption	1) Aggregate data are vulnerable at the nodes. 2) Does not work well for honest-but-curious nodes.
End-to-End Encryption	1) Does not work well for non-linear functions.
KIPDA	1) Provides a type of $k$ -indistinguishability.

# Privacy: Encryption vs. KIPDA

Method	Limitations
Hob-by-hop Encryption	<ol style="list-style-type: none"><li>1) Aggregate data are vulnerable at the nodes.</li><li>2) Does not work well for honest-but-curious nodes.</li></ol>
End-to-End Encryption	<ol style="list-style-type: none"><li>1) Does not work well for non-linear functions.</li></ol>
KIPDA	<ol style="list-style-type: none"><li>1) Provides a type of <math>k</math>-indistinguishability.</li><li>2) Secrets are in plain text but camouflaged.</li></ol>

# Privacy: Encryption vs. KIPDA

Method	Limitations
Hob-by-hop Encryption	<ol style="list-style-type: none"><li>1) Aggregate data are vulnerable at the nodes.</li><li>2) Does not work well for honest-but-curious nodes.</li></ol>
End-to-End Encryption	<ol style="list-style-type: none"><li>1) Does not work well for non-linear functions.</li></ol>
KIPDA	<ol style="list-style-type: none"><li>1) Provides a type of <math>k</math>-indistinguishability.</li><li>2) Secrets are in plain text but camouflaged.</li><li>3) Works well for honest-but-curious nodes.</li></ol>

# On the optimal sizes of sets

# On the optimal sizes of sets

- Sets sizes are determined in the following order:

# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:



# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:
    - A higher size gives more privacy.

# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:
    - A higher size gives more privacy.
    - A lower size uses less energy.

# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:
    - A higher size gives more privacy.
    - A lower size uses less energy.
  - 2 The restricted sets:

# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:
    - A higher size gives more privacy.
    - A lower size uses less energy.
  - 2 The restricted sets:
    - A higher size gives robustness to node-collusion.

# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:
    - A higher size gives more privacy.
    - A lower size uses less energy.
  - 2 The restricted sets:
    - A higher size gives robustness to node-collusion.
    - A lower size gives a higher  $k$  for  $k$ -indistinguishability.

# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:
    - A higher size gives more privacy.
    - A lower size uses less energy.
  - 2 The restricted sets:
    - A higher size gives robustness to node-collusion.
    - A lower size gives a higher  $k$  for  $k$ -indistinguishability.
  - 3 The global secret set:

# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:
    - A higher size gives more privacy.
    - A lower size uses less energy.
  - 2 The restricted sets:
    - A higher size gives robustness to node-collusion.
    - A lower size gives a higher  $k$  for  $k$ -indistinguishability.
  - 3 The global secret set:
    - Determined from the message and restricted set sizes. We give equations in the paper.

# On the optimal sizes of sets

- Sets sizes are determined in the following order:
  - 1 The message sets:
    - A higher size gives more privacy.
    - A lower size uses less energy.
  - 2 The restricted sets:
    - A higher size gives robustness to node-collusion.
    - A lower size gives a higher  $k$  for  $k$ -indistinguishability.
  - 3 The global secret set:
    - Determined from the message and restricted set sizes. We give equations in the paper.
- The reverse order determines the size of the message set given the required minimal amount of node collusion.



# Challenges to KIDPA

# Challenges to KIDPA

- Nodes that are more than honest-but-curious, and will subvert the network aggregates.

# Challenges to KIDPA

- Nodes that are more than honest-but-curious, and will subvert the network aggregates.
- Not as efficient with streaming encryption techniques.

# Challenges to KIDPA

- Nodes that are more than honest-but-curious, and will subvert the network aggregates.
- Not as efficient with streaming encryption techniques.
- Information is not 100% concealed, only indistinguishable.

# Challenges to KIDPA

- Nodes that are more than honest-but-curious, and will subvert the network aggregates.
- Not as efficient with streaming encryption techniques.
- Information is not 100% concealed, only indistinguishable.
- Still need to exchange the restricted sets with the nodes and the base station every often.

# Conclusion

# Conclusion

- First work we are aware of that provides “indistinguishability” to privacy preserving data aggregation.

# Conclusion

- First work we are aware of that provides “indistinguishability” to privacy preserving data aggregation.
- Saves energy and time even though more messages are sent.



# Future Work

# Future Work

- Implement in TOSSIM or similar WSN simulator.

# Future Work

- Implement in TOSSIM or similar WSN simulator.
- Address other adversarial models.

# Future Work

- Implement in TOSSIM or similar WSN simulator.
- Address other adversarial models.
  - Byzantine attacks.

# Future Work

- Implement in TOSSIM or similar WSN simulator.
- Address other adversarial models.
  - Byzantine attacks.
  - Denial-of-Service attacks.

# Future Work

- Implement in TOSSIM or similar WSN simulator.
- Address other adversarial models.
  - Byzantine attacks.
  - Denial-of-Service attacks.
  - Node insertion attacks.

# Future Work

- Implement in TOSSIM or similar WSN simulator.
- Address other adversarial models.
  - Byzantine attacks.
  - Denial-of-Service attacks.
  - Node insertion attacks.
- Address mobility in nodes.

Thank you for your attention.  
Questions?