Unassailable Sensor Networks

Alessandro Panconesi, La Sapienza

Joint work with:

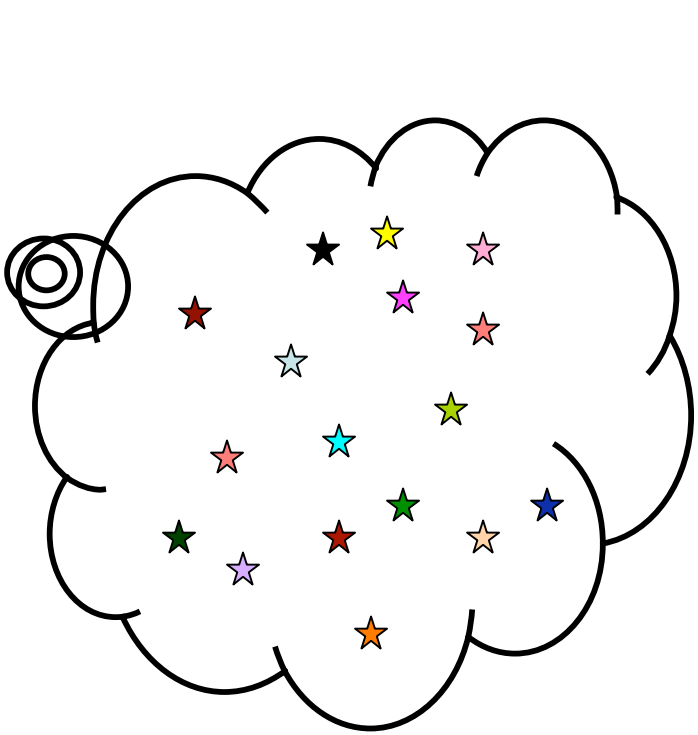
Luigi V. Mancini, Alessandro Mei, La Sapienza

Roberto DiPietro, Roma Tre

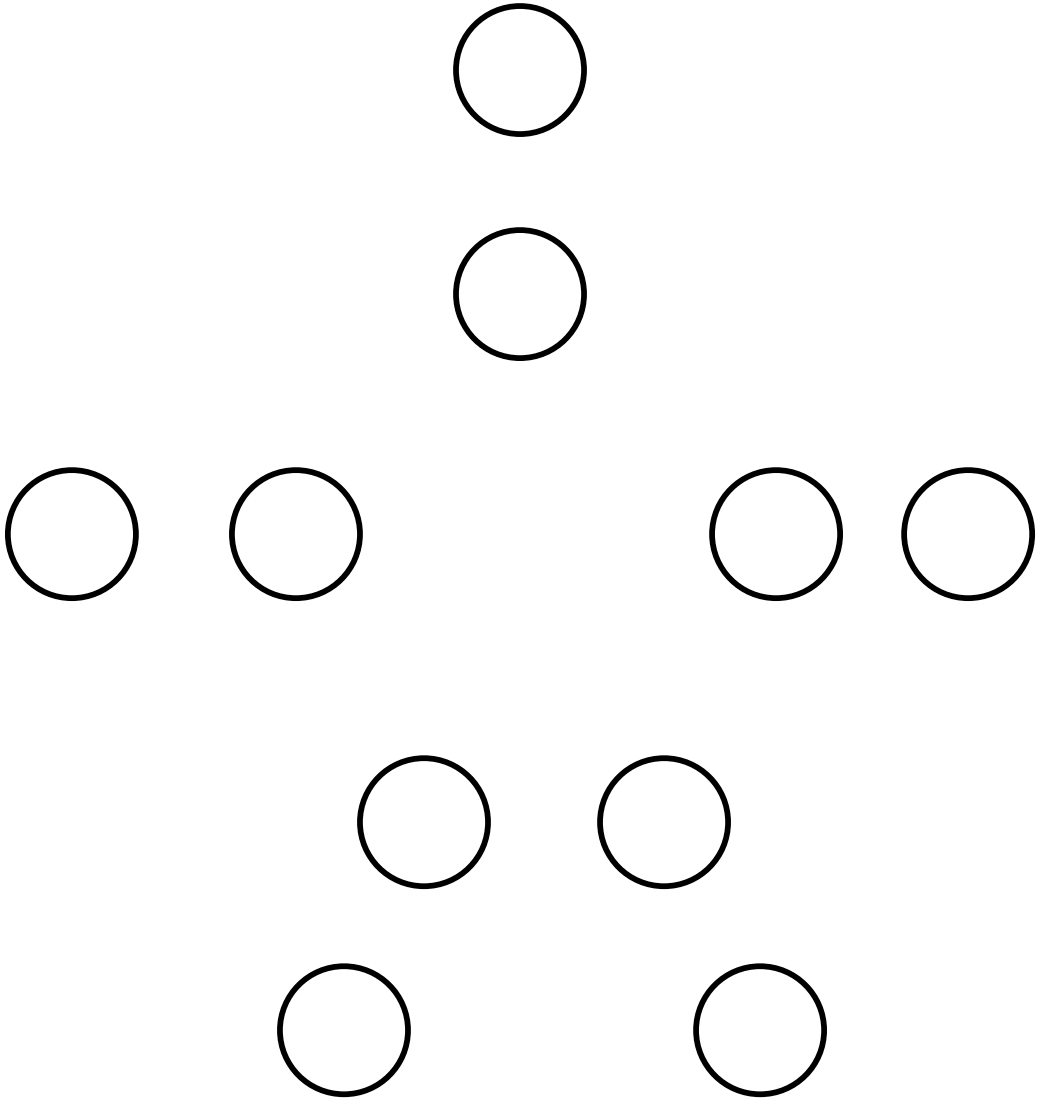
Jaikumar Radhakrishnan, Tata Institute of Fundamental Research

Secure sensor networks

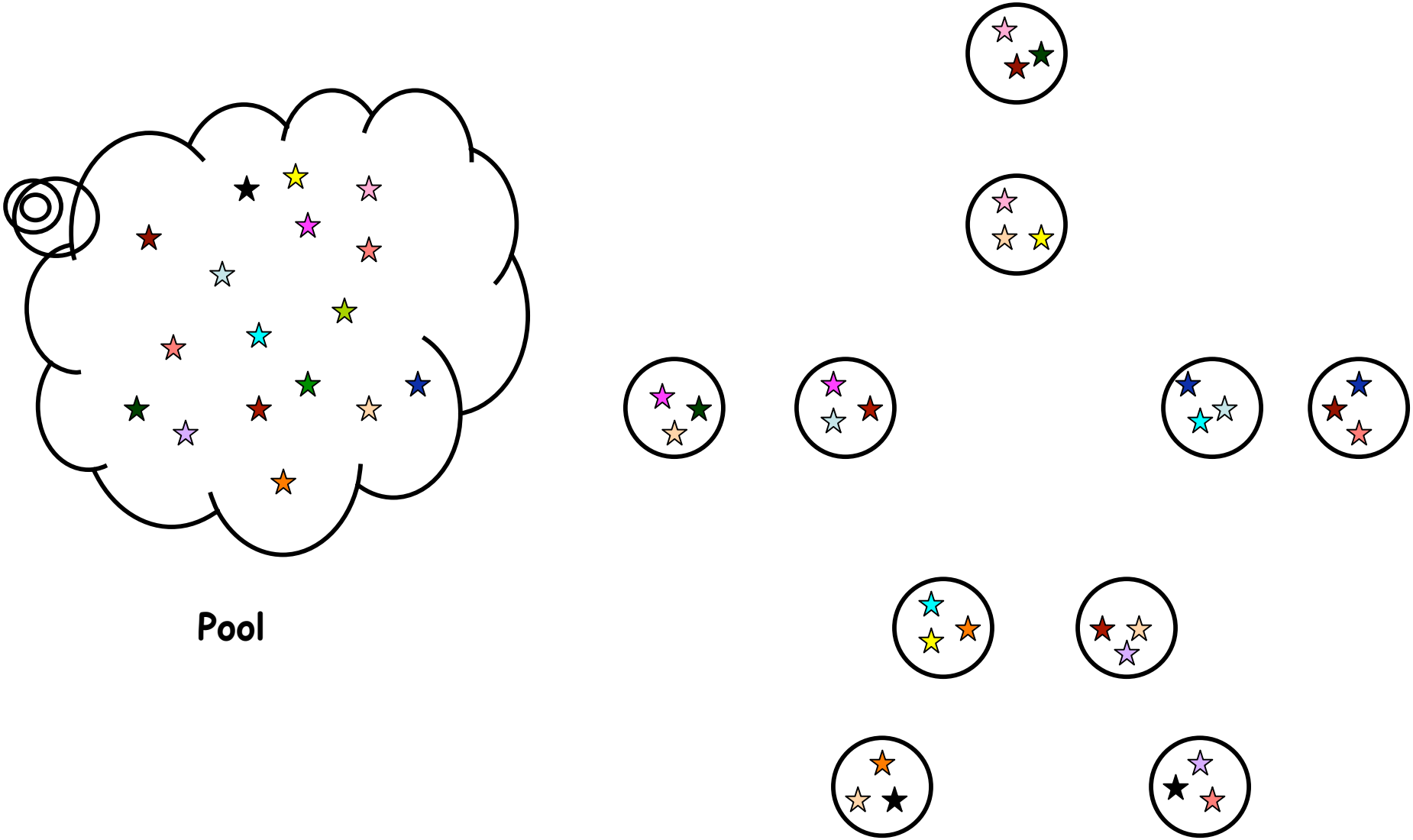
Random pre-distribution of keys



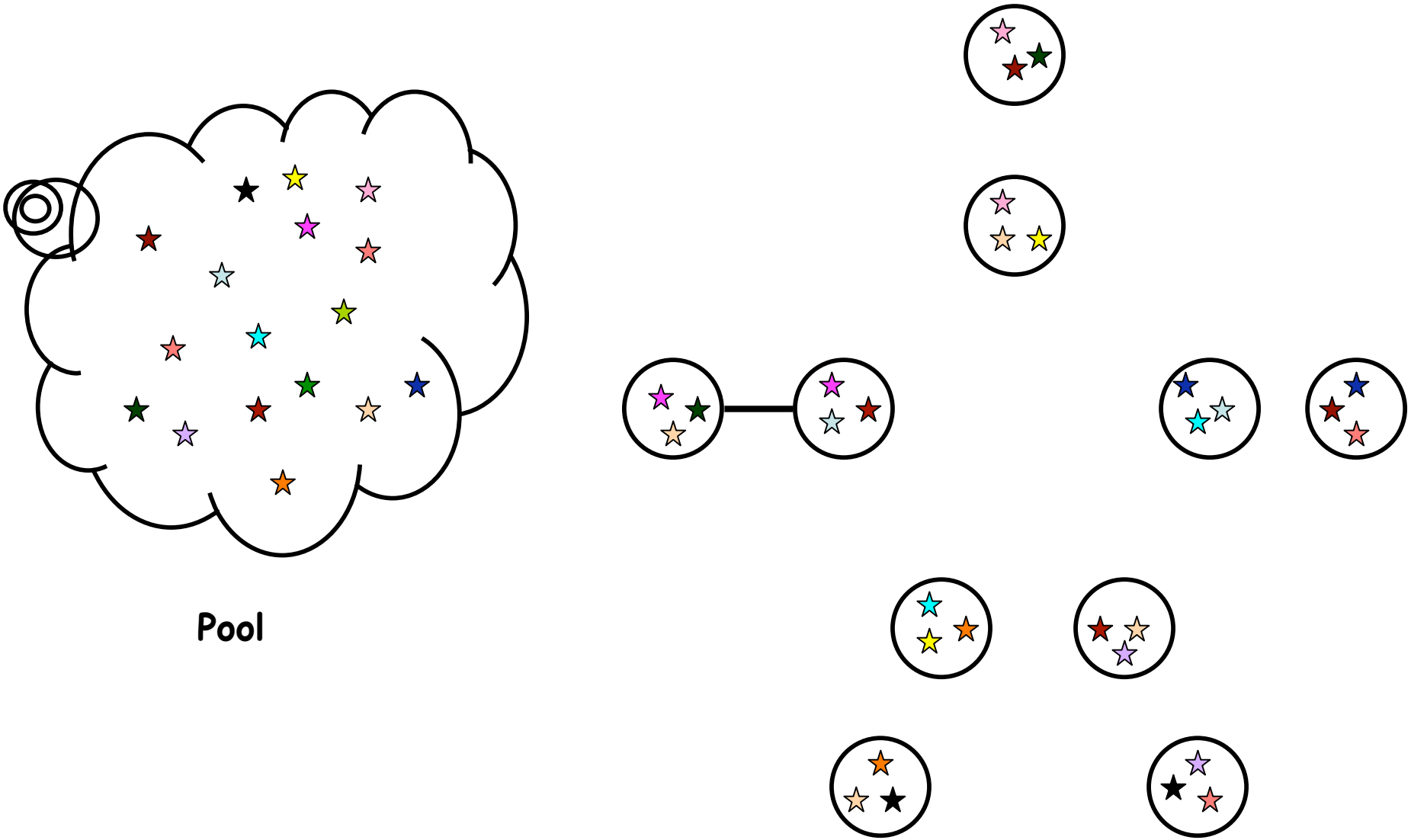
Pool



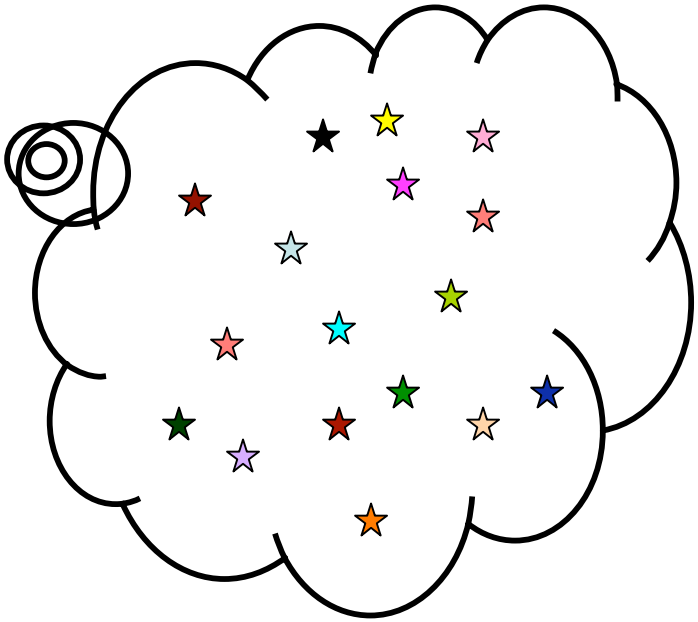
Random pre-distribution of keys



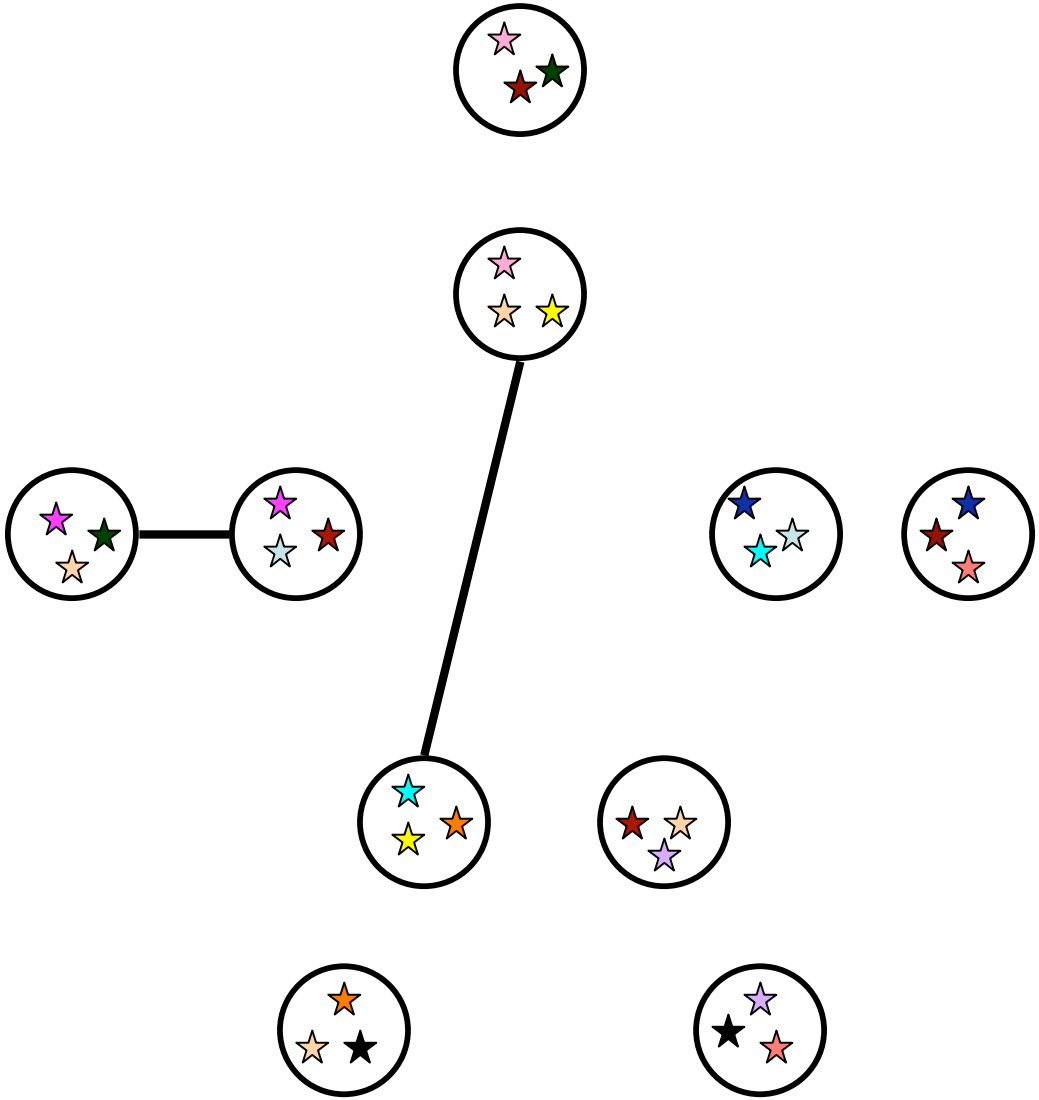
Random pre-distribution of keys



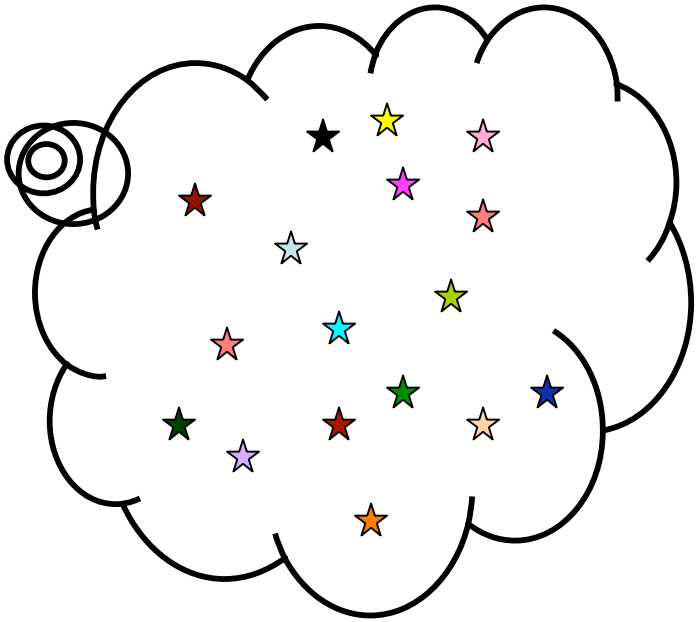
Random pre-distribution of keys



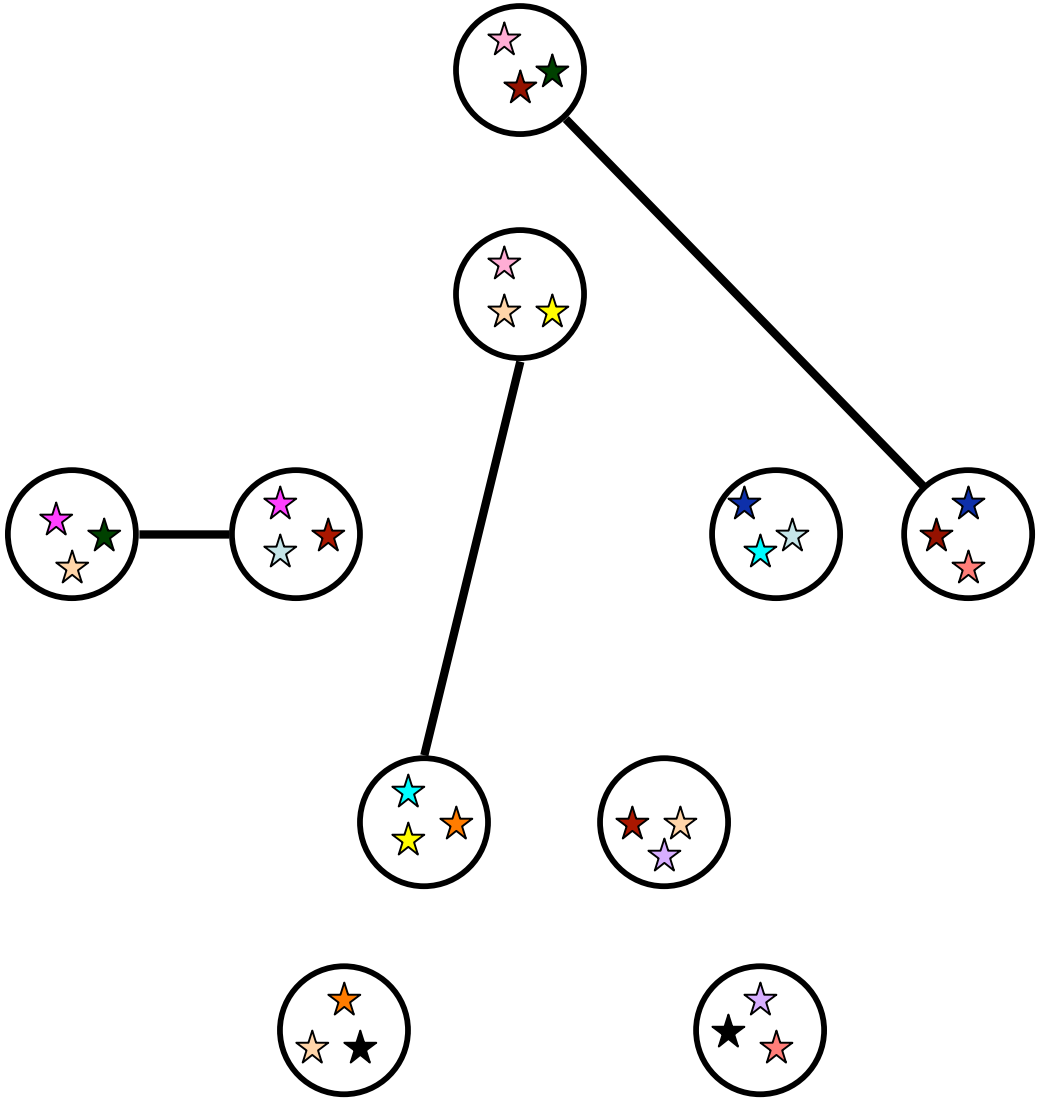
Pool



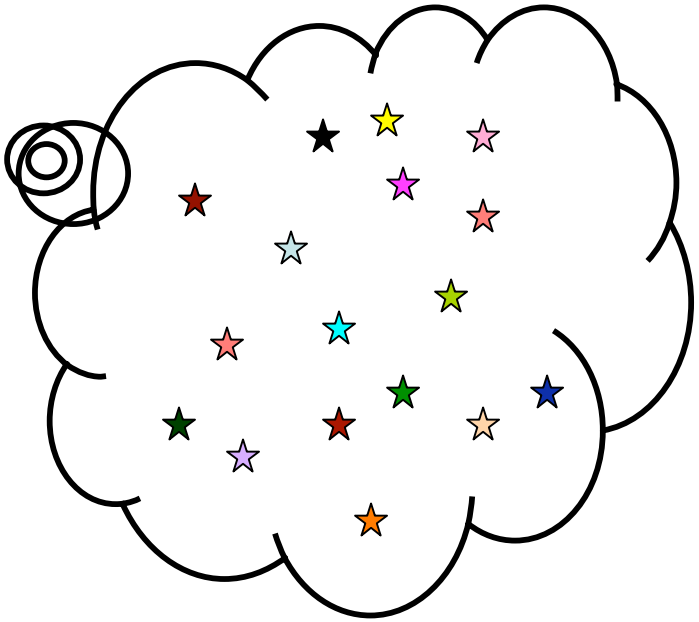
Random pre-distribution of keys



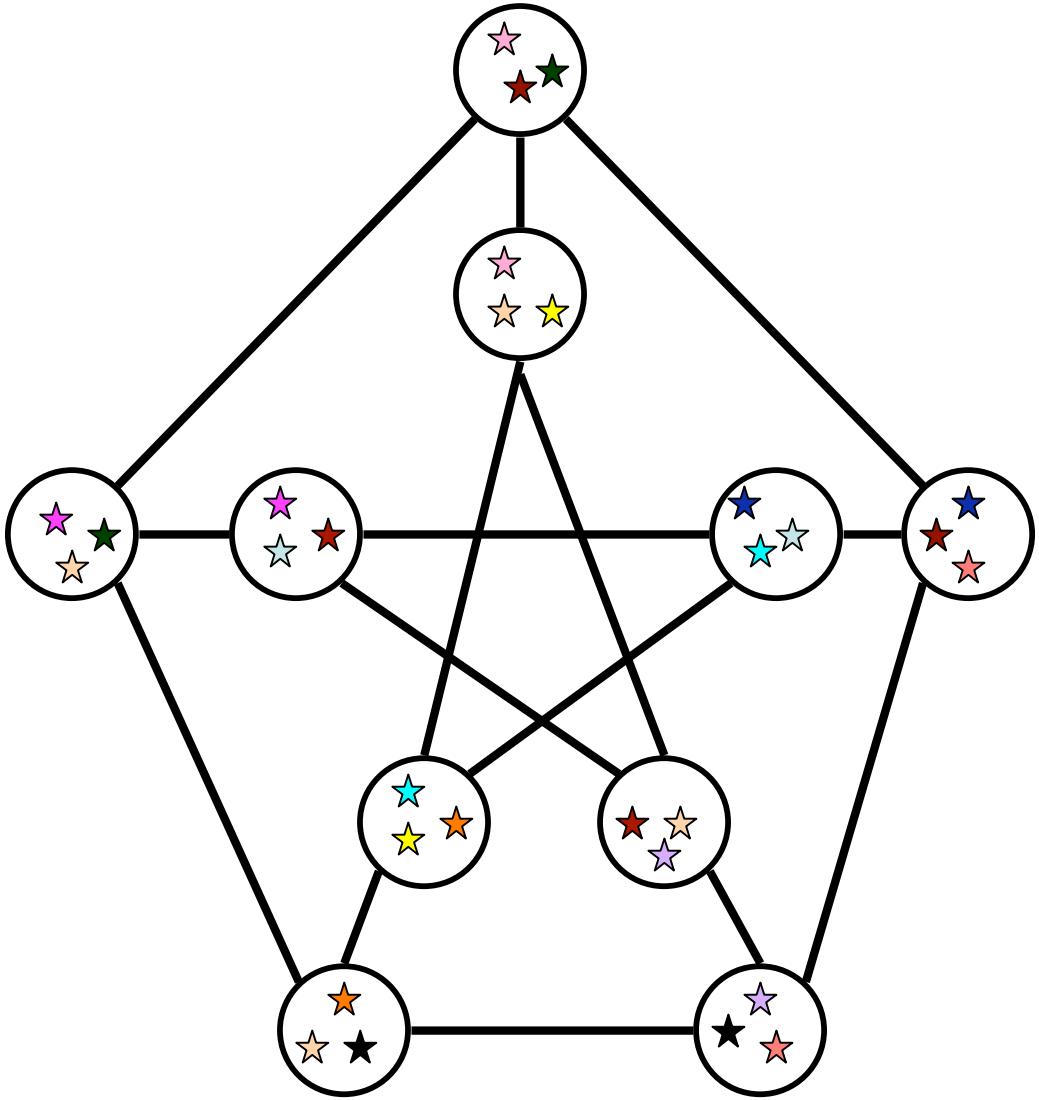
Pool



Random pre-distribution of keys



Pool



Aims of this talk

- To convince Jean-Pierre that there is still a lot of interesting work to do in this framework

- Give a warning: Model is not only not completely understood, it is in fact misunderstood

Kryptographs

There are N nodes (or vertices):

- Each vertex u is given a key ring of size k, sampled at random from a pool of size K

- Two vertices u and v are neighbours iff they share a key

The resulting graph is called a kryptograph G(N,k,K)

Geometric Kryptographs

- N nodes (vertices) are distributed at random within the unit square

- Each vertex u is given a key ring of size k, sampled at random from a pool of size K

- Two vertices u and v are neighbours iff they share a key and are within transmission range r

The resulting graph is called a (geometric) kryptograph G(N,r,k,K)

Random pre-distribution of keys

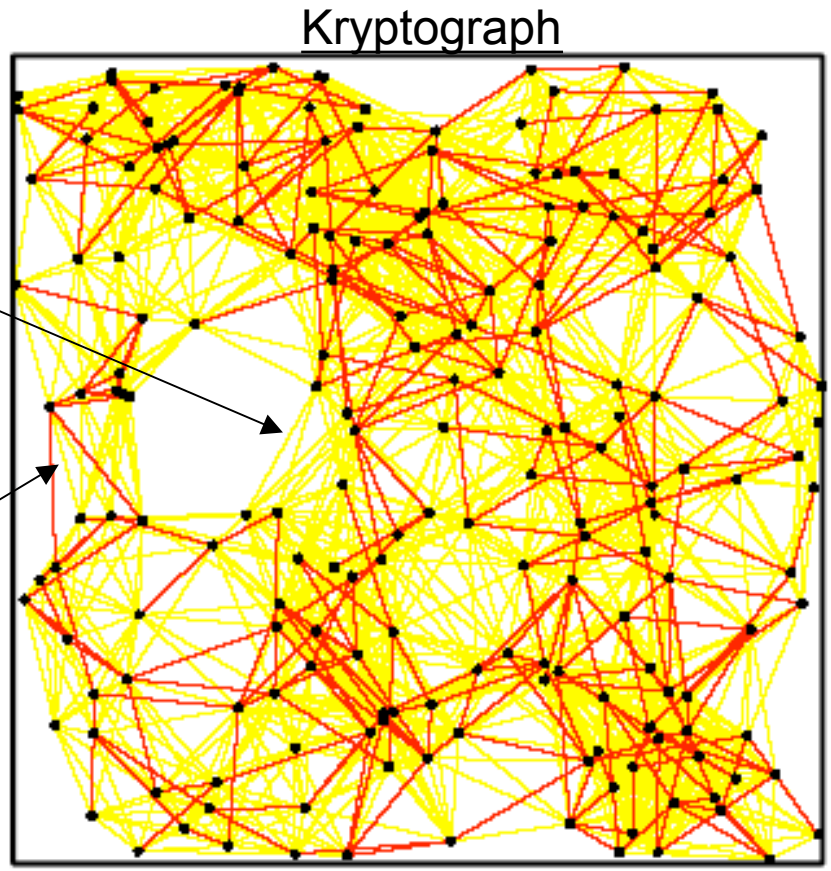
We can set up a WSN that it is connected via encryption-protected links

Random pre-distribution of keys

We can set up a WSN that it is connected via encryption-protected links

Yellow link: two sensors are in the same communication range, **BUT** they do not share a key

Red link: two sensors are in the same communication Range, **AND** they do share at least a key



Random pre-distribution of keys

Originally introduced for sensor networks this elegant idea is general enough to apply to other scenarios, e.g. peer-to-peer networks

As discussed during this course, there are many interesting developing or new scenarios.

Keep your eyes open!

Our Goal

Given N (number of nodes) and r (transmission range), fix key-ring size k and pool-size K in such a way that the network is at the same time

1. Connected
2. Secure against massive attacks

We need to define precisely what we mean by "security". We'll do so in a minute

Conflicting goals

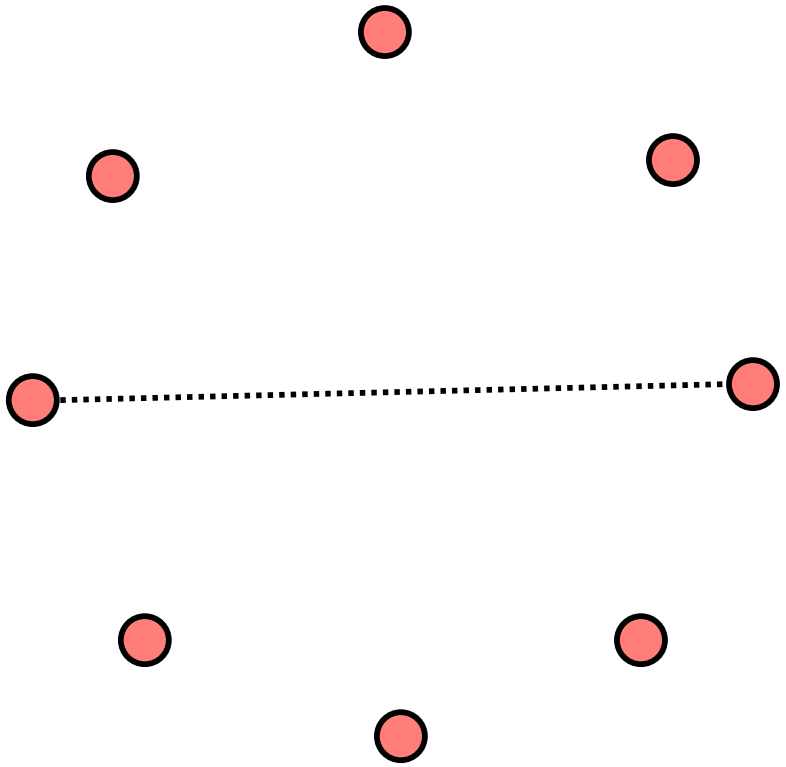
Note that connectivity and security are at odds:

1. Connectivity wants the key ring to be large, since this increases the probability of having links
2. Security likes them to be small, since in this fashion capturing a key is unlikely to corrupt many links

We will show that k and K can be fixed to have BOTH. To the best of our knowledge this approach is entirely new

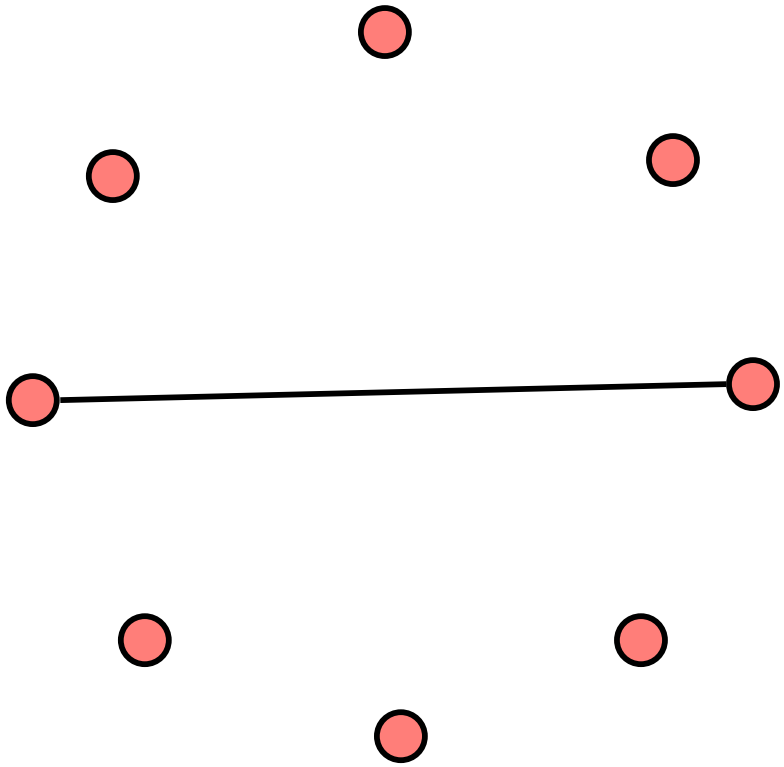
Misunderstandings

Erdös-Renyi



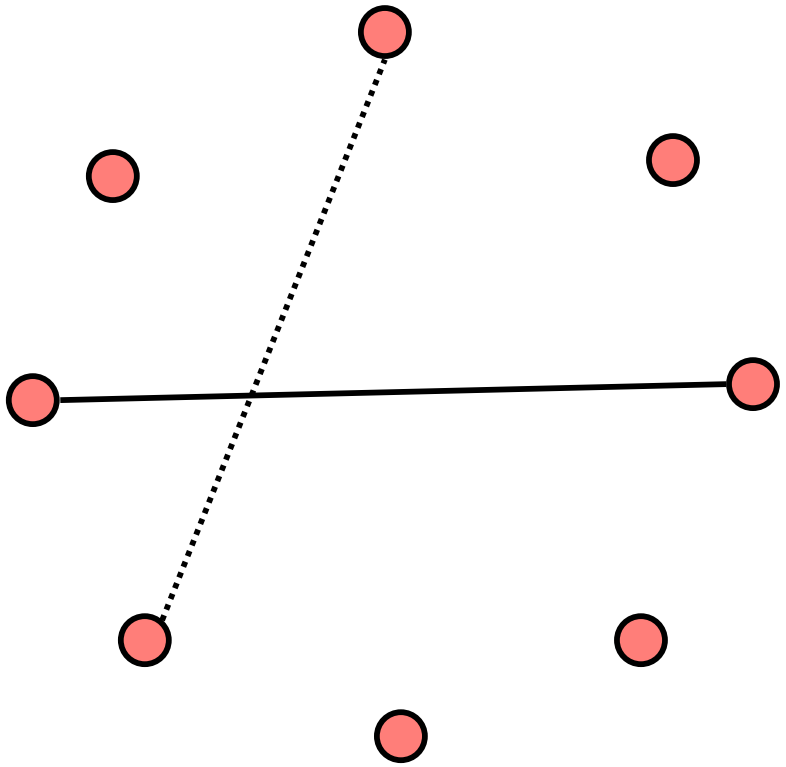
N = #vertices p = edge probability

Erdös-Renyi



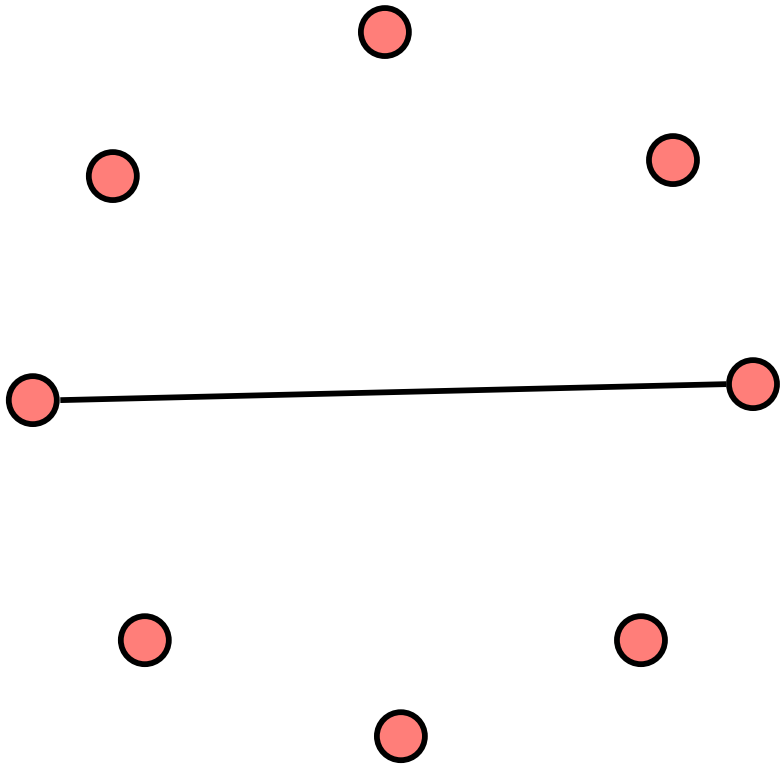
N = #vertices p = (key ring)^2/(pool size)

Erdös-Renyi



N = #vertices p = (key ring)^2/(pool size)

Erdös-Renyi

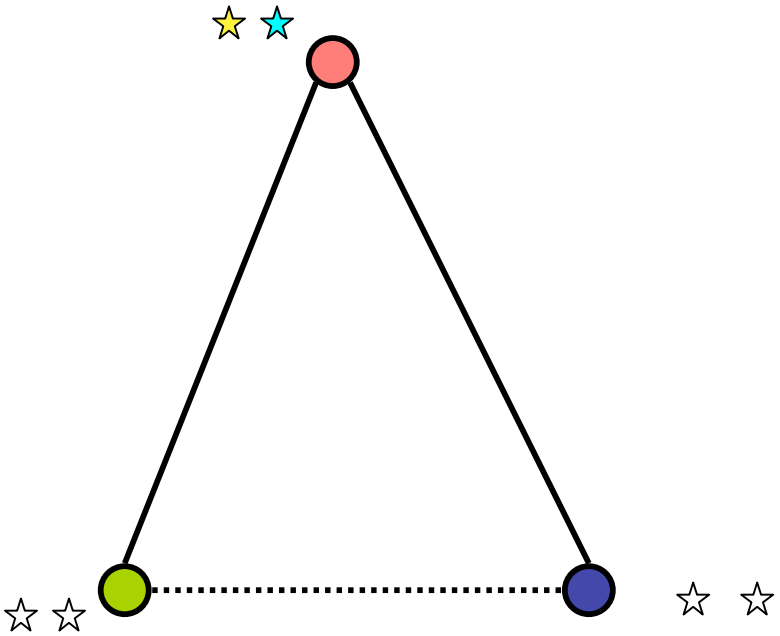


N = #vertices p = (key ring)^2/(pool size)

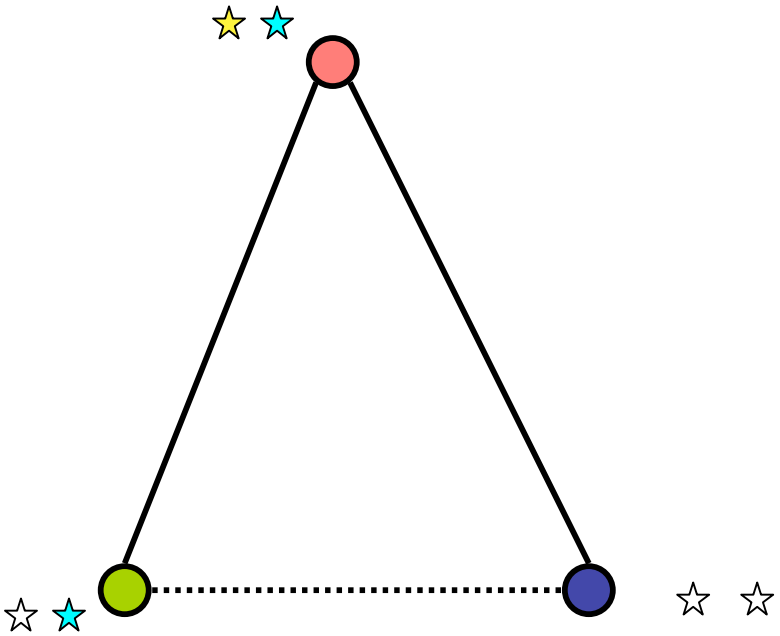
Erdös-Renyi

Crucially, edges exist independently of each other

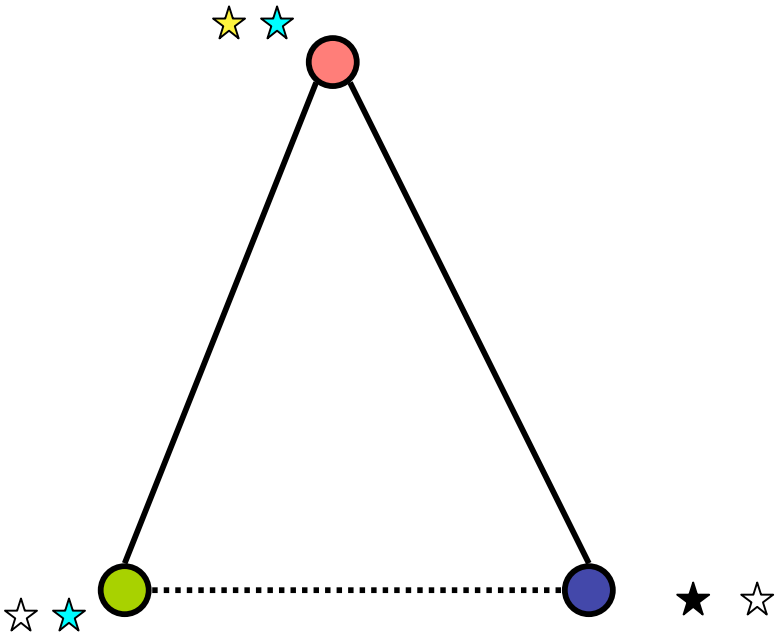
Triangles



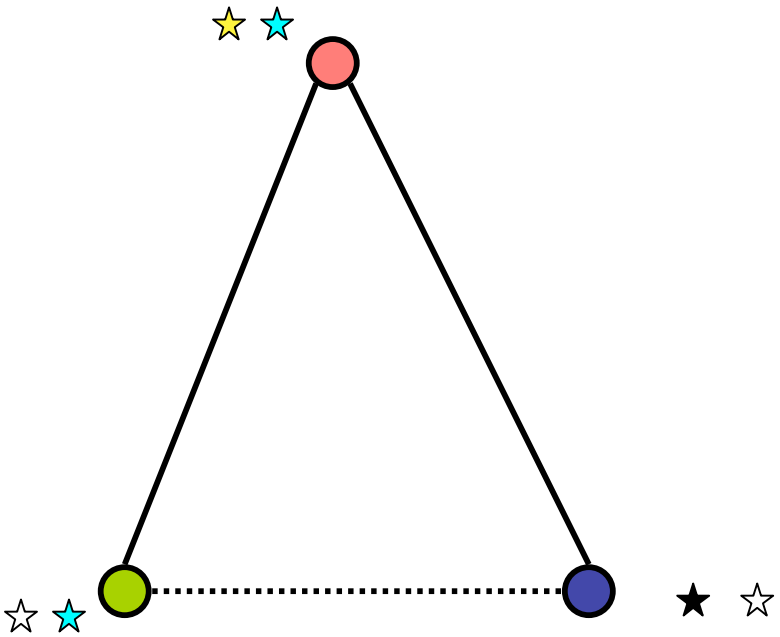
Triangles



Triangles

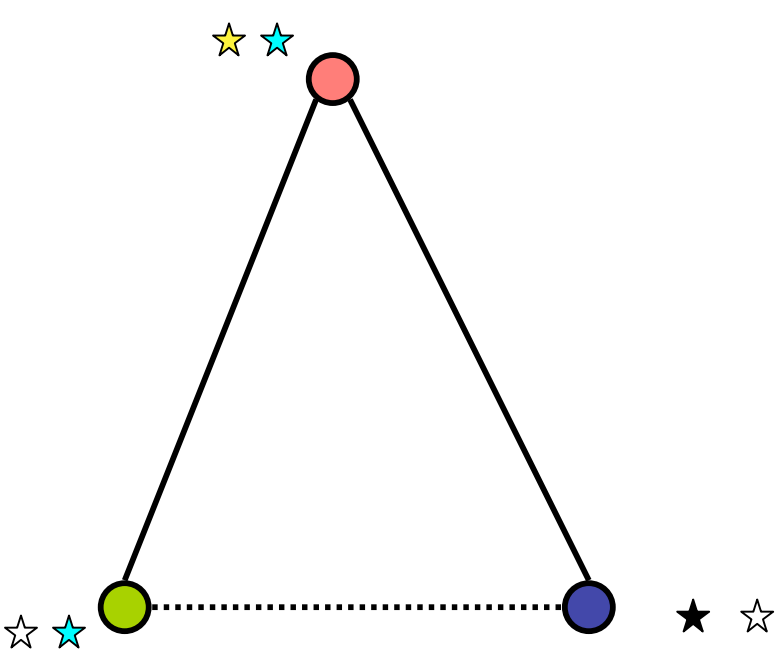


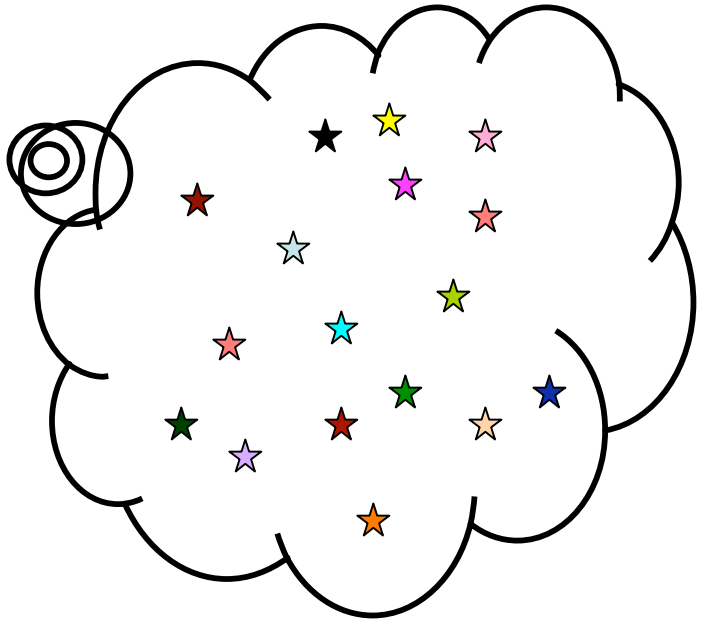
Triangles



Now ☆ must be either ☆ or ☆: Pr(edge exists) > 1/2

Triangles



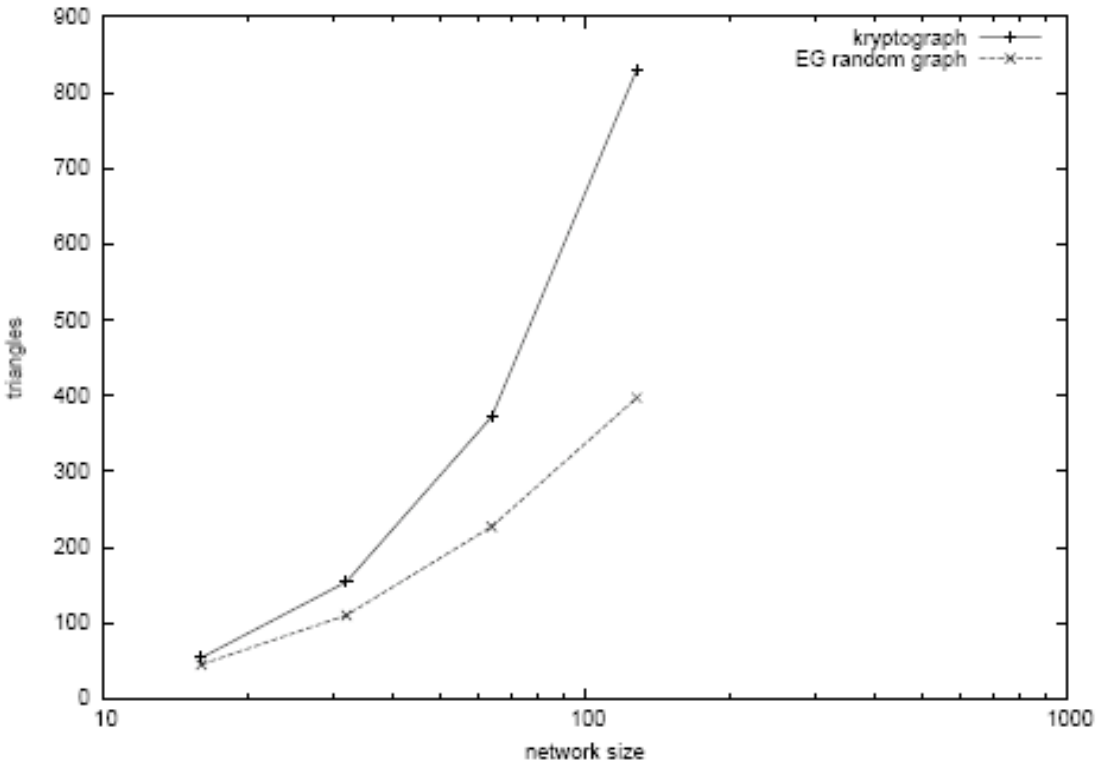


Pool

Kryptograph : Pr(edge exists) > 1/2

Erdös-Renyi random graph: Pr(edge exists) = 2/(pool size) = 0.00……001

Structural differences



Methodological remarks

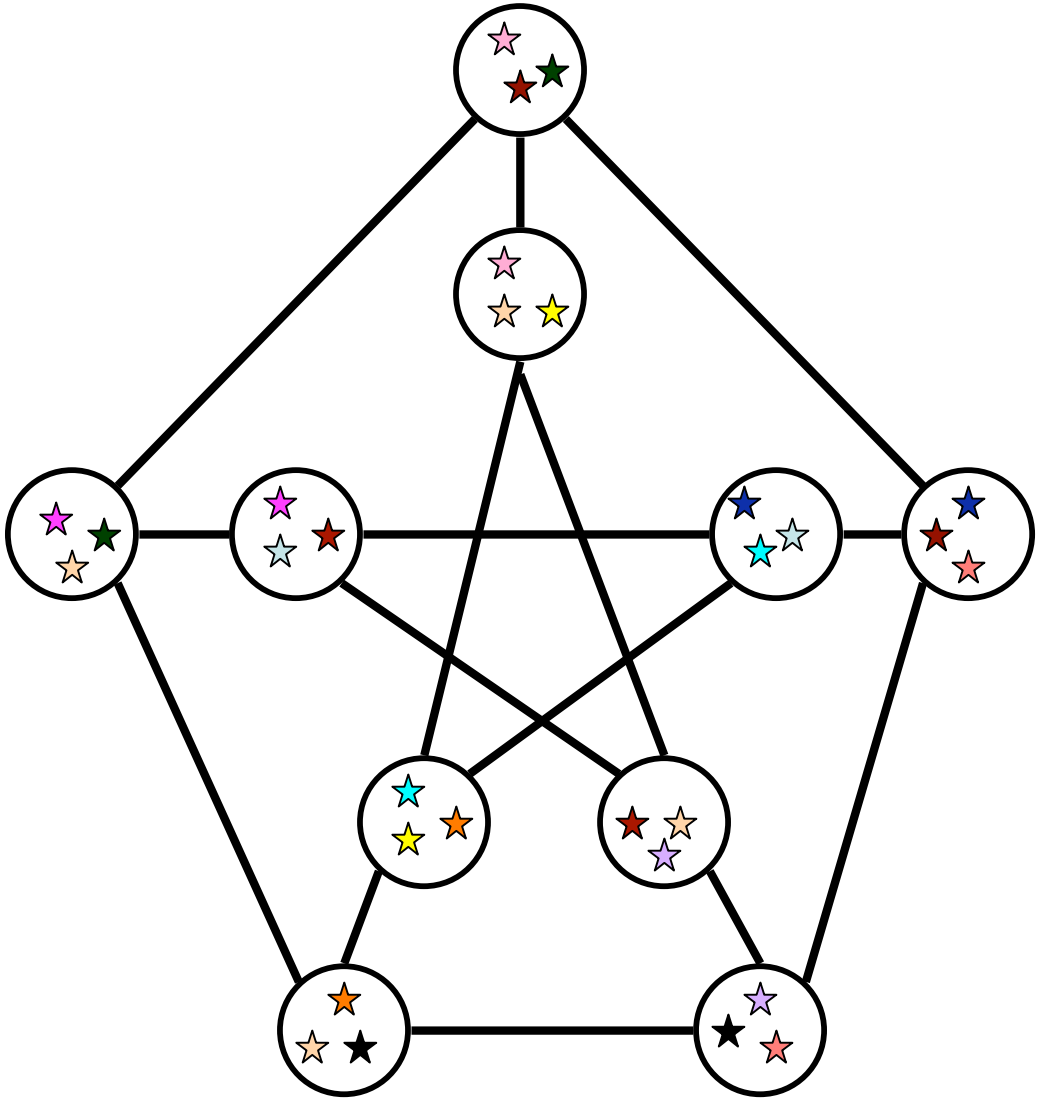
1. No need to approximate with Erdös-Renyi model, kryptographs can be analized exactly
2. Precise understanding is always important, but especially so when SECURITY is at stake
3. Indeed, note that "triangle" dependency above is BAD for security: if adversary owns key ring it is more likely than with Erdös-Renyi to corrupt links between neighbours of captured node
4. Also note: unclear how to formally define security properties in Erdös-Renyi framework. Indeed, security approached only experimentally in the literature

Our notion of security

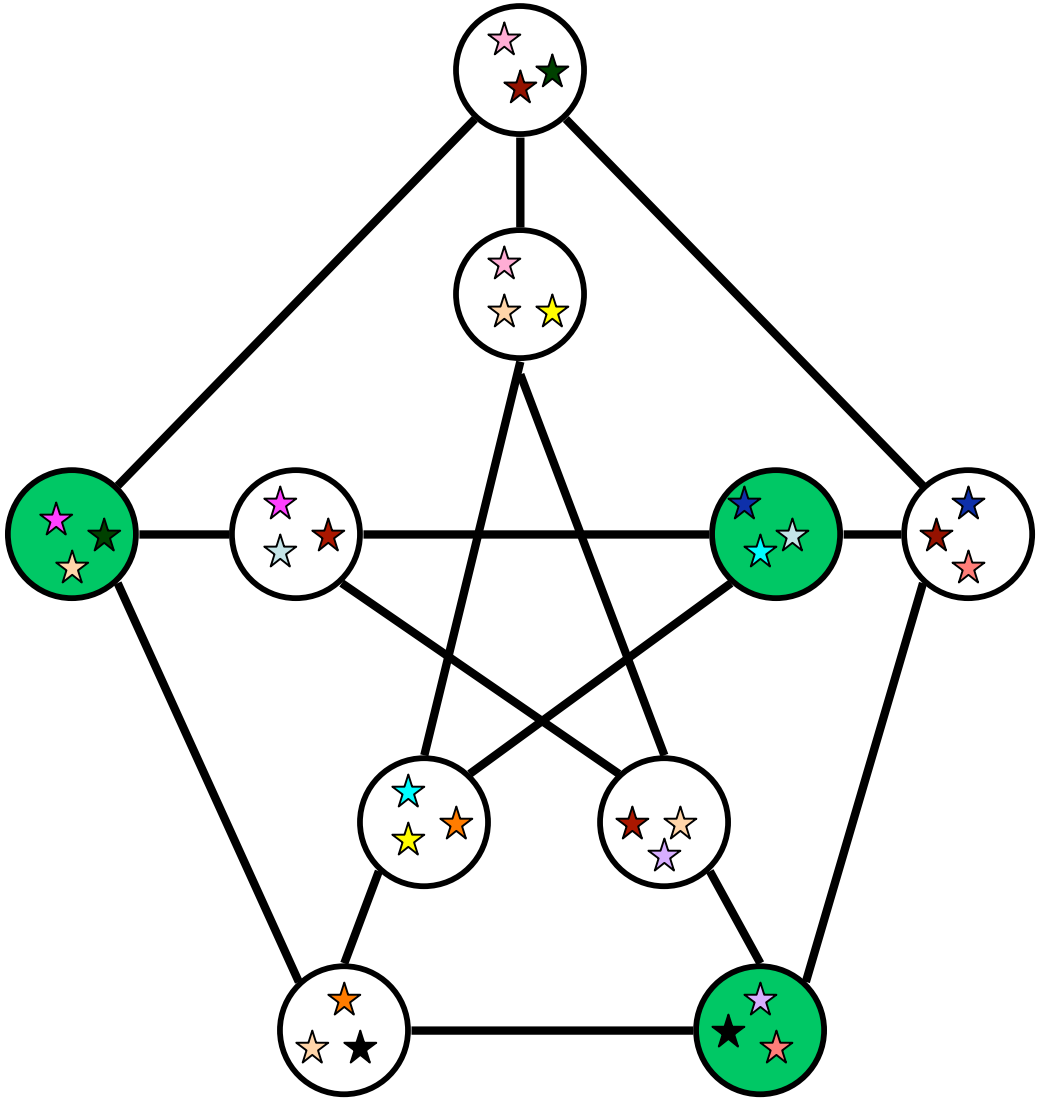
Suppose that by capturing a tiny fraction of the nodes (e.g. $\sqrt{n}$ nodes) we can compromise a linear fraction of the links. We would consider the network highly insecure

Conversely, we would consider it secure if, in order to compromise a large fraction of the links, a large fraction of the nodes must be captured

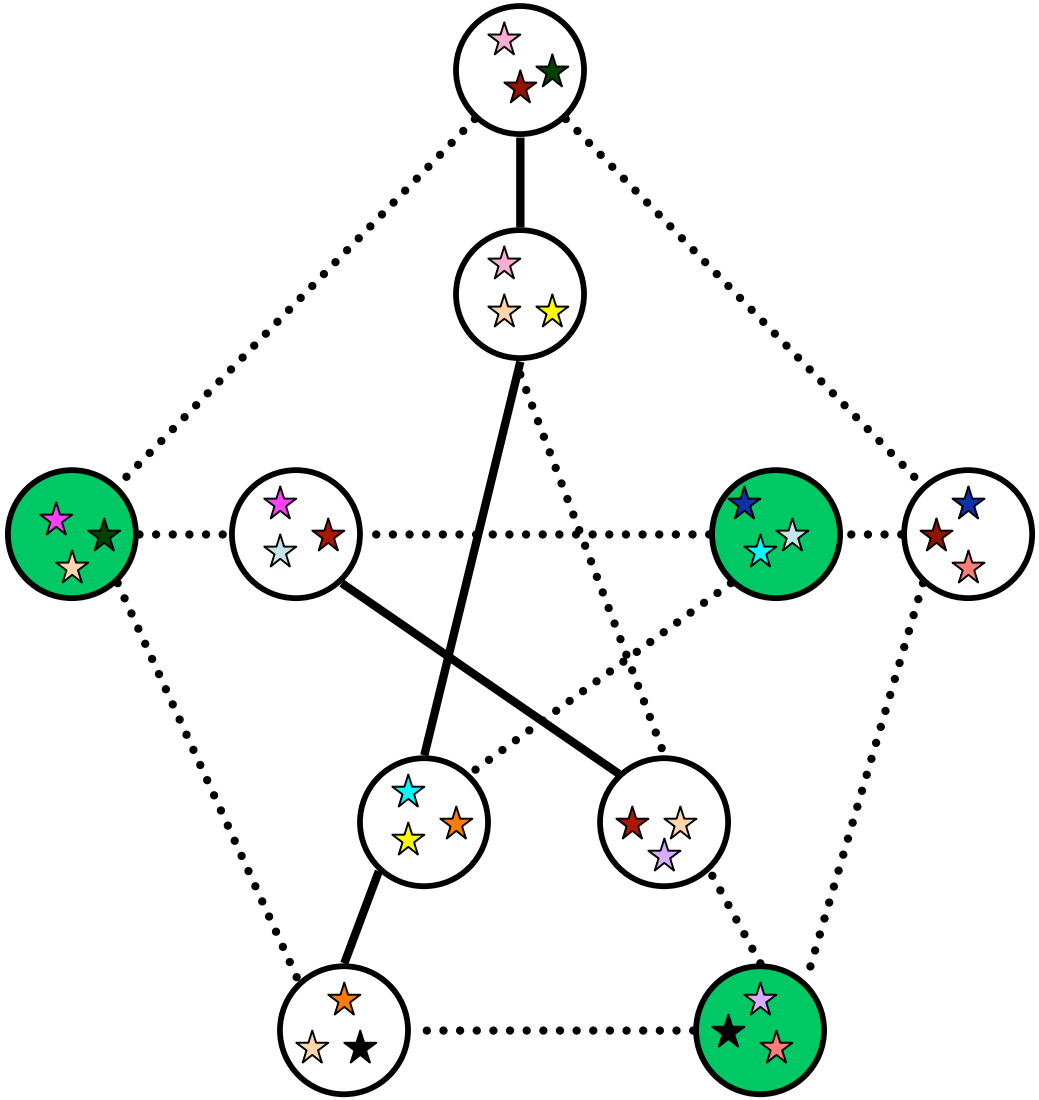
Security against Massive Attacks



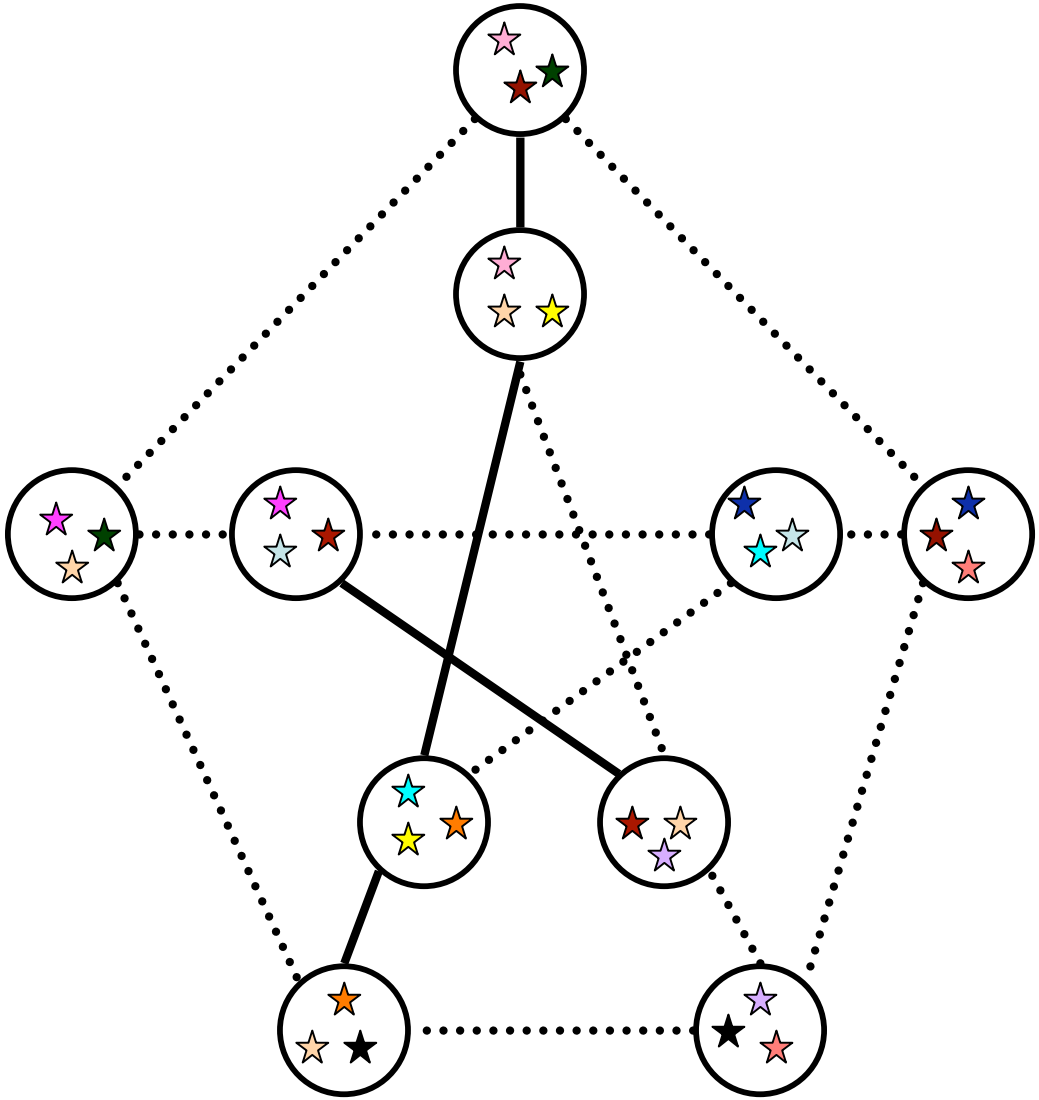
Security against Massive Attacks



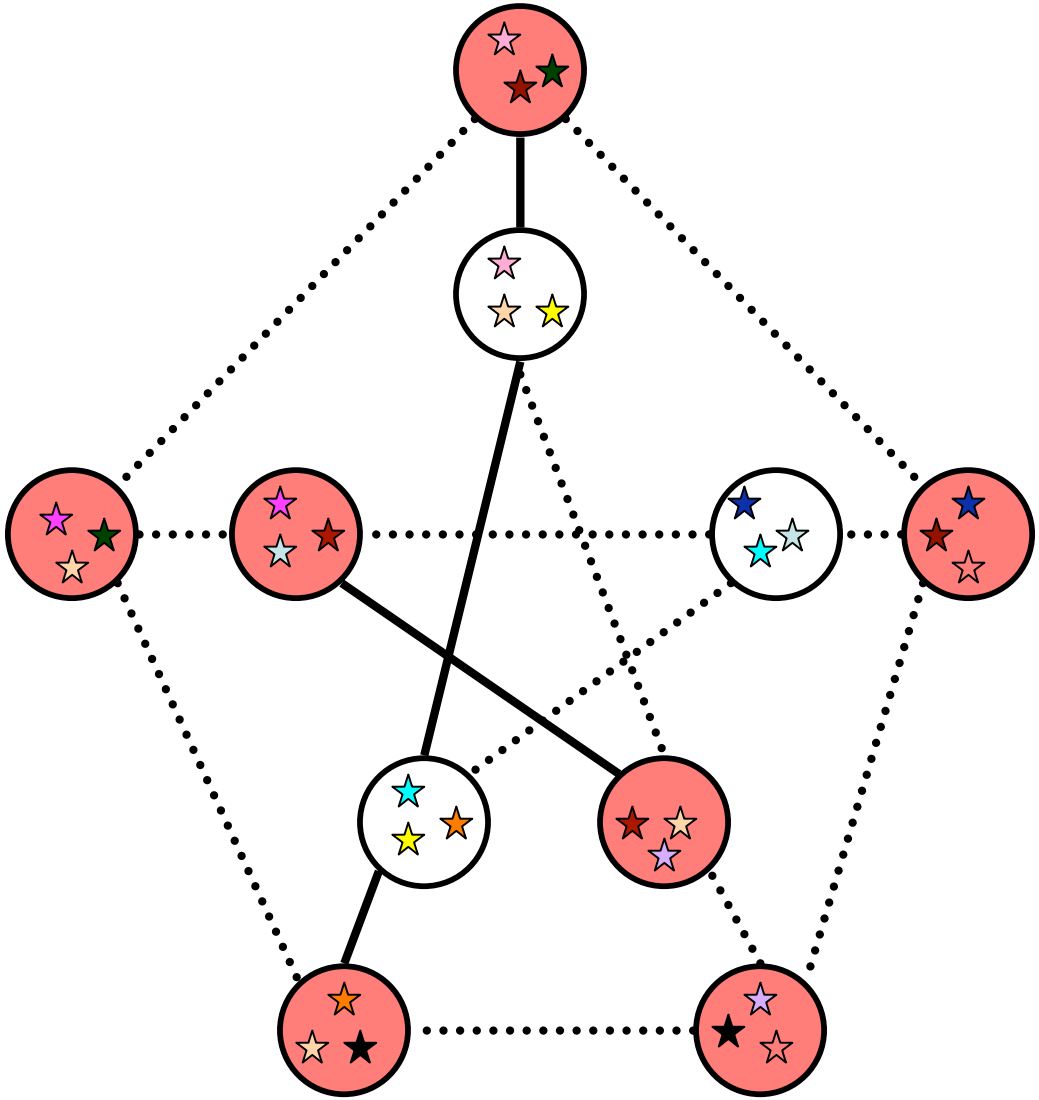
Security against Massive Attacks



Security against Massive Attacks



Security against Massive Attacks



Redoubtable Networks

We consider massive attacks. The RANDOM ADVERSARY selects nodes at random:

✓ if a node is selected its entire key ring is owned by the adversary

✓ if uv is a link and the adversary owns a key in $K(u) \cap K(v)$ the link is compromised (note this is a pessimistic assumption since the key might not be used)

The aim of the adversary is to compromise linearly many links. We say that the network is REDOUBTABLE if to do so the adversary must select linearly many nodes

Our Goal (revisited)

Given N and r, fix k and K in such a way that the network is at the same time, with high probability

1. Connected
2. Redoubdtable

Main Theorem

If

$$(\text{key ring size})^2/(\text{pool size}) \sim \log(N)/N$$

Then, with high probability, the network is both connected and redoubdtable

High probability means that the probability that anything goes wrong goes (very quickly) to zero as N grows. For instance, assume N = 256, K = 16,384, and k = 128, then $\mathbf{p \approx 2^{-23}}$

Remark: if $k^2/K \ll \log(N)/N$ network is disconnected with good (constant) probability

Unassailable Networks

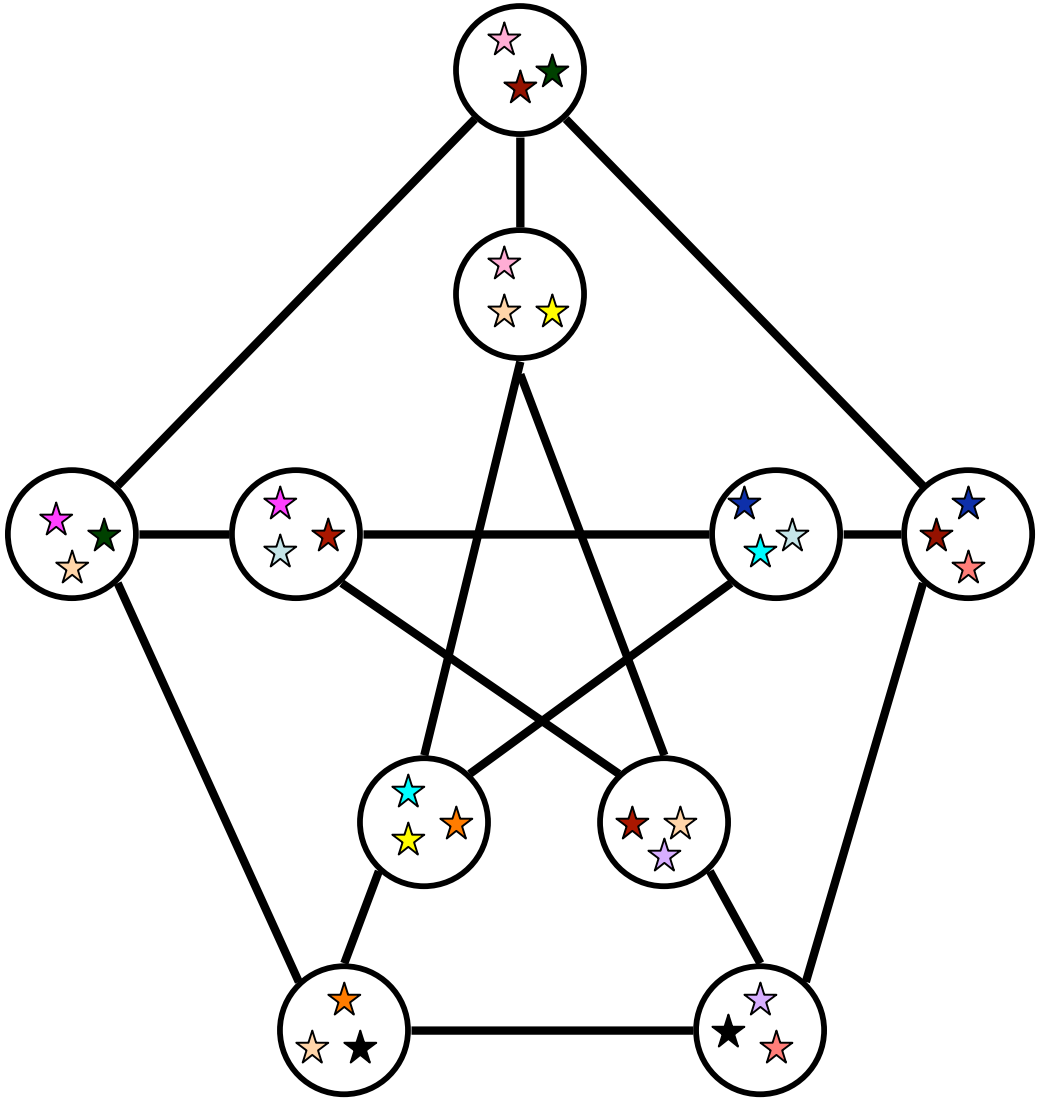
The OMNISCENT ADVERSARY:

✓ It knows the entire distribution of keys, I.e. it knows the sets K(u), for every vertex u
✓ As before, if a node is selected its entire key ring is owned by the adversary and a link uv is compromised as soon as it owns a key in $K(u) \cap K(v)$
✓ Its aim of the adversary is to compromise linearly many links.

We say that the network is UNASSAILABLE if to do so the omniscent adversary must select linearly many nodes

The Omniscent Adversary

The omniscent adversary knows how the keys are distributed. How many nodes does it have to capture to compromise a linear fraction of the links?



Our Goal

Given N and r, fix k and K in such a way that the network is at the same time, with high probability

1. Connected
2. Unassailable

Omniscent Adversary

Unrealistically strong?

Omniscent Adversary

Unrealistically strong? Yes, but this is entirely desirable!

- Certifying security against it, automatically subsumes weaker, more realistic attackers (e.g. random attackers)
- It shows light weight, insecure key-discovery protocols are not unsafe (against the kind of attack we are considering)

Main Theorem

If

$$(\text{key ring size})^2/(\text{pool size}) \sim \log(N)/N$$

Then, with high probability, the network is both connected and unassailable

Main Theorem

If

$$(\text{key ring size})^2/(\text{pool size}) \sim \log(N)/N$$

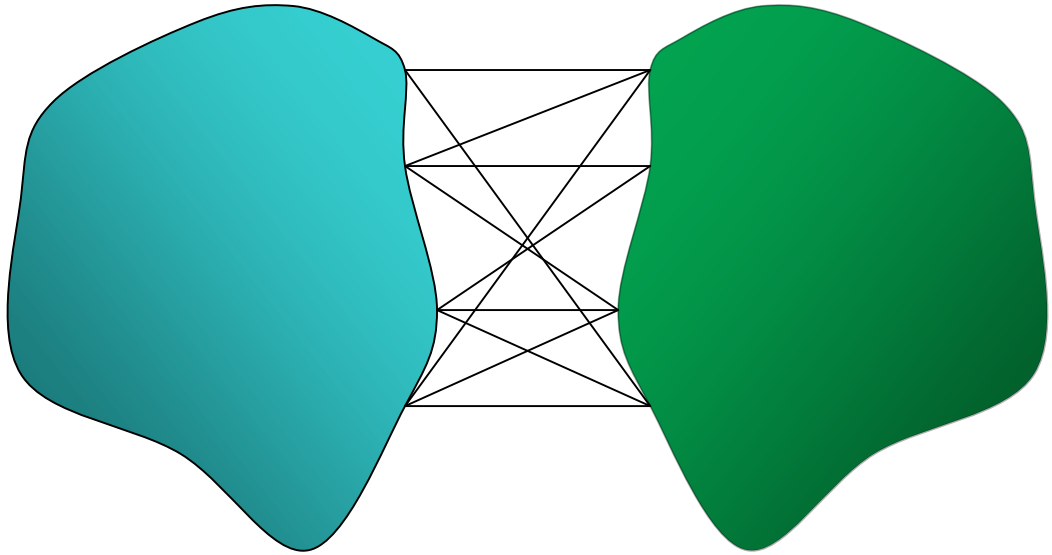
Then, with high probability, the network is both connected and unassailable

High probability means that the probability that anything goes wrong goes (very quickly) to zero as N grows. For instance, assume N = 256, K = 16,384, and k = 128, then $\mathbf{p \approx 2^{-23}}$

Remark: if $k^2/K \ll \log(N)/N$ network is disconnected with good (constant) probability

Unsplittable Networks

Suppose now that the adversary wants to split the network into two large chunks, compromising all links between them (I.e. it wants to partition the network). Can it do it by capturing a small set of nodes?



2nd Main Theorem

If

$$(\text{key ring size})^2/(\text{pool size}) \sim \log(N)/N$$

Then, with high probability, the network is both connected and unsplittable (and also unassailable)

3rd Main Theorem

If

$$(\text{key ring size})^2/(\text{pool size}) \sim 1/N$$

Then, with high probability, the network has a GIANT COMPONENT that is at the same time unsplittable and unassailable

3rd Main Theorem

If

$$(\text{key ring size})^2/(\text{pool size}) \sim 1/N$$

Then, with high probability, the network has a GIANT COMPONENT that is at the same time unsplittable and unassailable

Small key ring size gives several benefits in resource starving environments

A giant component is sparse (degree is constant instead of log(N)). And yet it has very strong connectivity properties: It is an EXPANDER

4th Main Theorem

We also know that the kryptograph and the giant component are EXPANDERS

To summarize

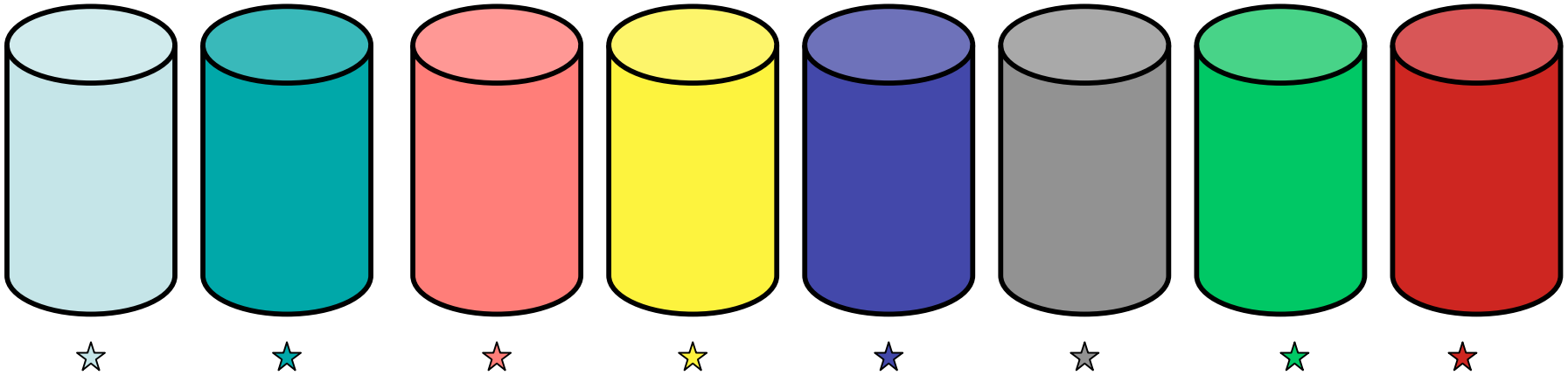
Kryptographs have very strong connectivity, security and fault-tolerant properties:

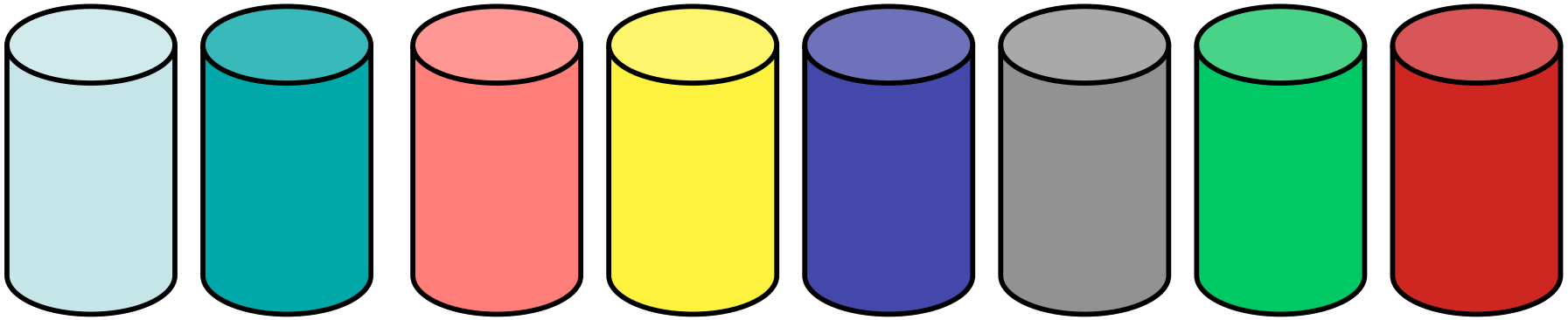
- If $k^2/K \sim \log(N)/N$ they are, with high probability, connected, unsplittable and unassailable. Furthermore they are expanders

- If $k^2/K \sim 1/N$ they have, with high probability, a giant component that is unsplittable, unassailable and (we bet..) is an expander

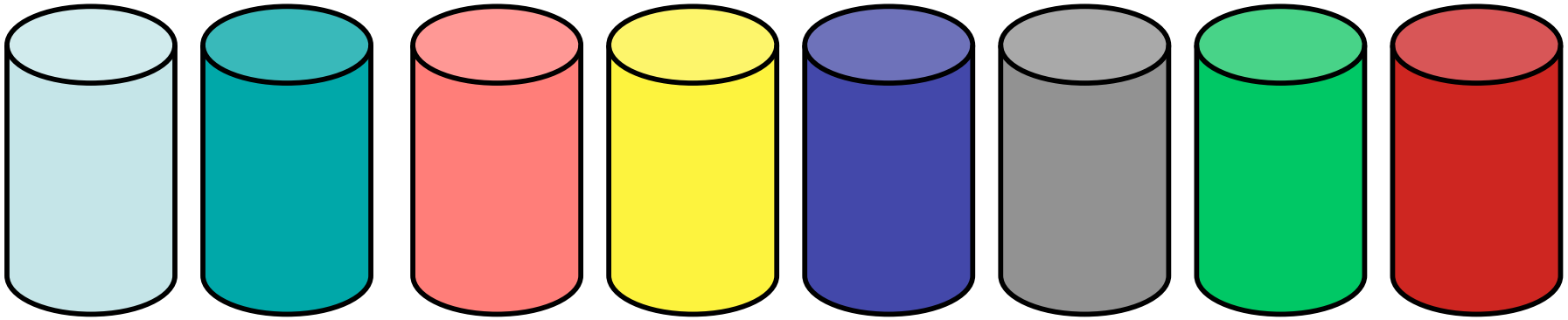
Analysis:

Framing the omniscent adversary





1 2 3 4 5 6 7 8 9

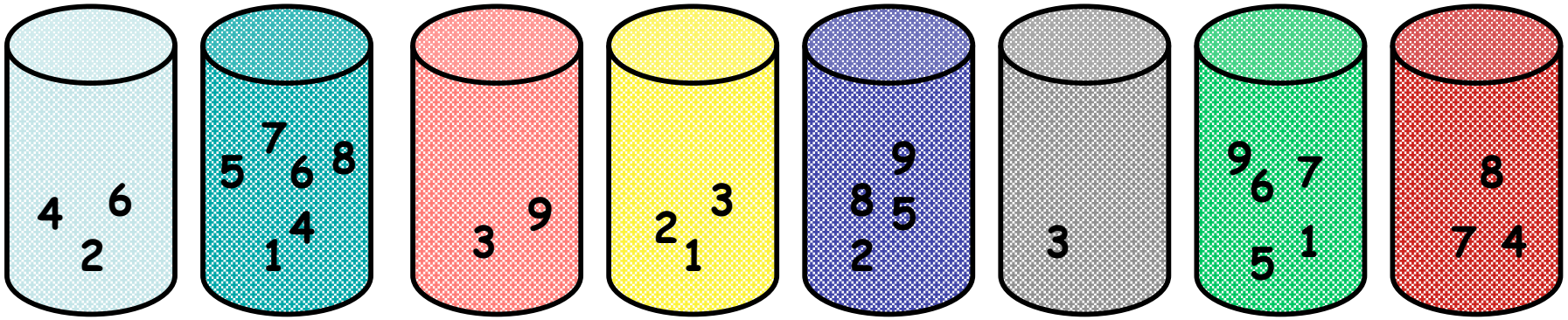


1 2 3 4 5 6 7 8 9

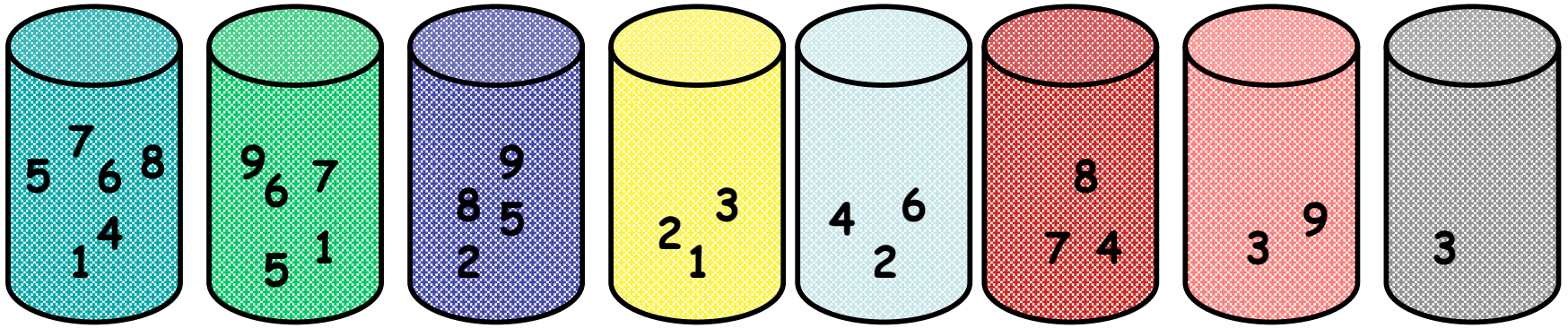
1 2 3 4 5 6 7 8 9

1 2 3 4 5 6 7 8 9

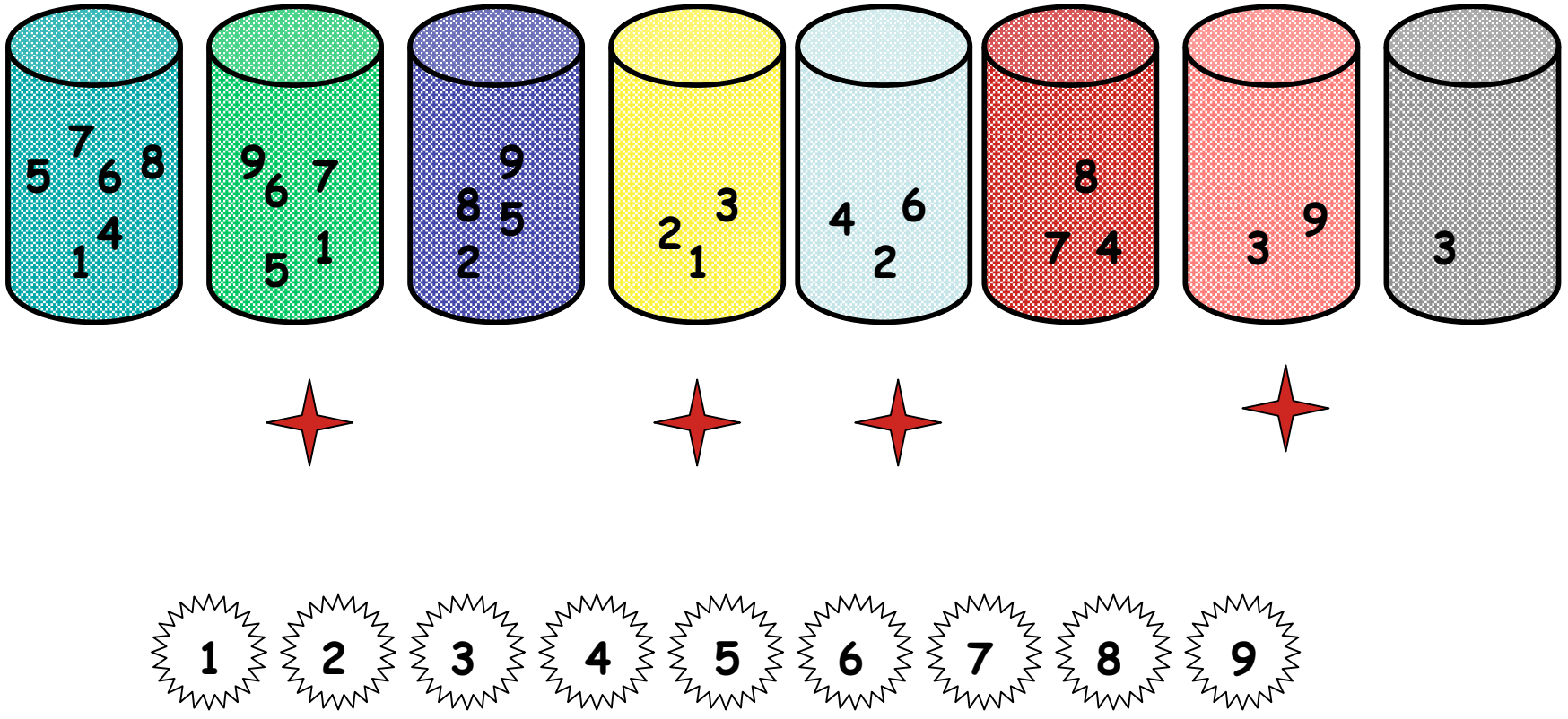
1 2 3 4 5 6 7 8 9

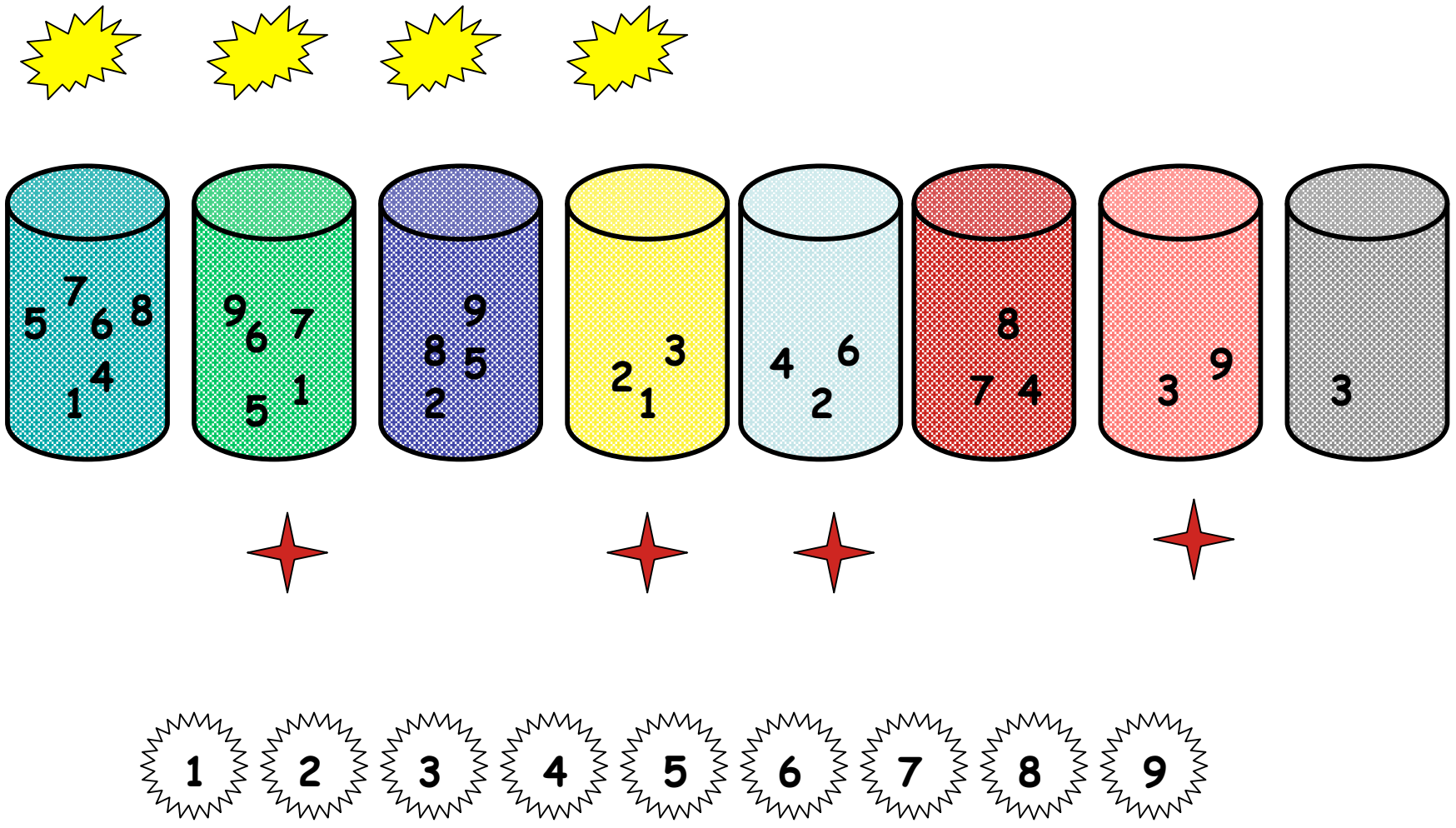

4 6 2
7 5 6 8 1 4
3 9
2 3 1
9 8 5 2
3
9 6 7 5 1
8 7 4


1
2
3
4
5
6
7
8
9



1 2 3 4 5 6 7 8 9


7
5 6 8
4
1
9
6 7
1
5
9
8 5
2
3
2 1
6
4 2
8
7 4
9
3
3
1
2
3
4
5
6
7
8
9

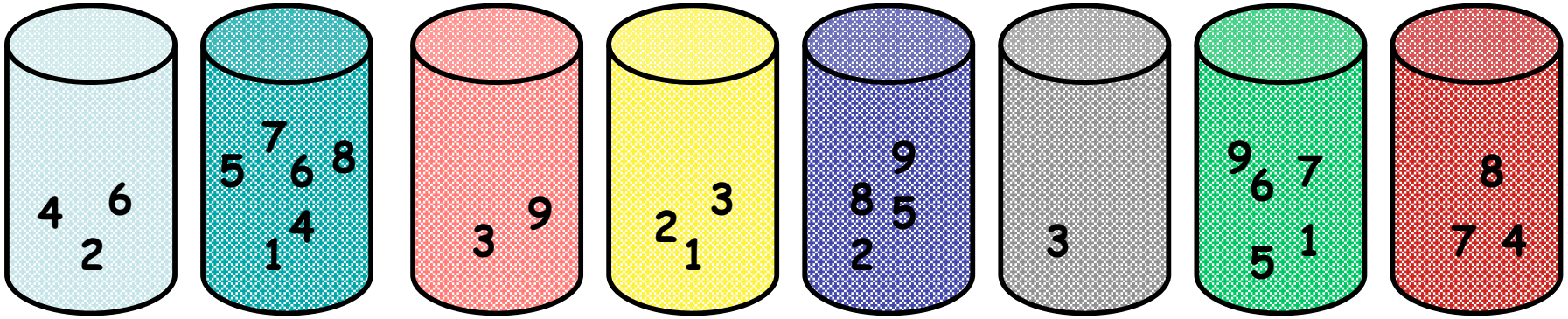

7
5 6 8
4
1
9
6 7
1
5
9
8 5
2
3
2 1
6
4 2
8
7 4
9
3
3

1
2
3
4
5
6
7
8
9

- FACT 1: if the adversary picks t bins, the maximum damage is given by the t largest bins

- FACT 1: if the adversary picks t bins, the maximum damage is given by the t largest bins
- FACT 2: if $K \geq N^2$ then, with high probability, no bin has more than 5 balls I.e. no key can compromise more than 10 edges

Useful bins



- FACT 1: if the adversary picks t bins, the maximum damage is given by the t largest bins
- FACT 2: if $K \geq N^2$ then, with high probability, no bins has more than 5 balls I.e. no key can compromise more than 10 edges
- FACT 3: With high probability, no vertex has more than log(N) useful bins

- FACT 1: if the adversary picks t bins, the maximum damage is given by the t largest bins
- FACT 2: if $K \geq N^2$ then, with high probability, no bins has more than 5 balls I.e. no key can compromise more than 10 edges
- FACT 3: With high probability, no vertex has more than log(N) useful bins
- FACT 4: WHP, the graph has N log(N) edges

Proof

- The adversary must compromise $\Theta(N \log(N))$ edges
- It needs $\Theta(N \log(N))$ useful bins, since each bin can yield at most 10 edges
- Each node can contribute at most log(N) useful bins
- Hence it must capture $\Theta(N)$ nodes

QED

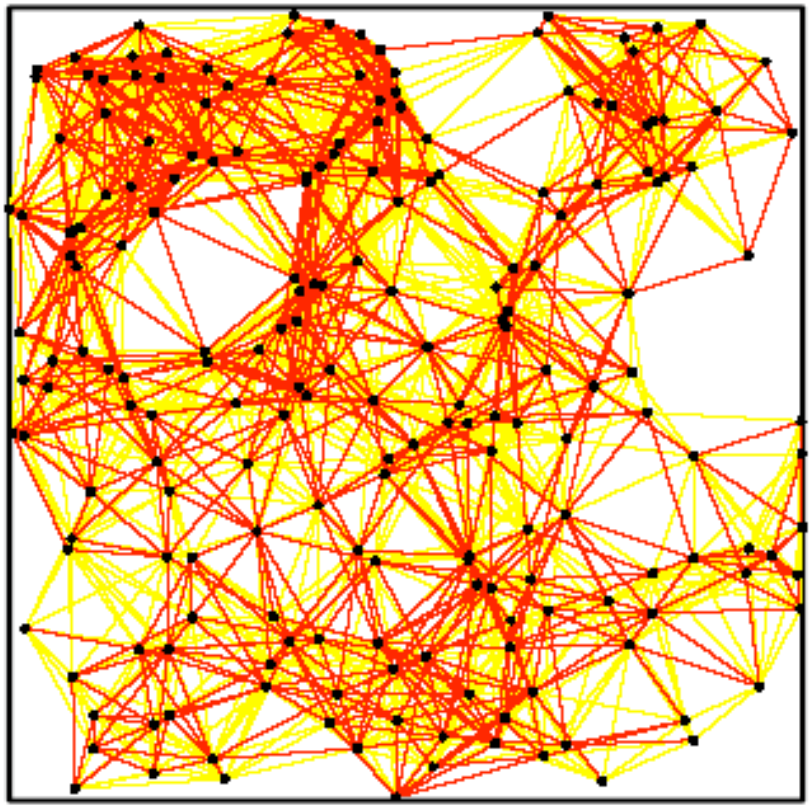
Tools

- Doing the analysis requires large deviation inequalities
- Assuming $K \geq N^2$ simplifies the analysis considerably. Chernoff-Hoeffding is enough
- Assuming $K \geq N \log(N)$ requires a more interesting proof, and more sophisticated Martingale arguments

Experiments

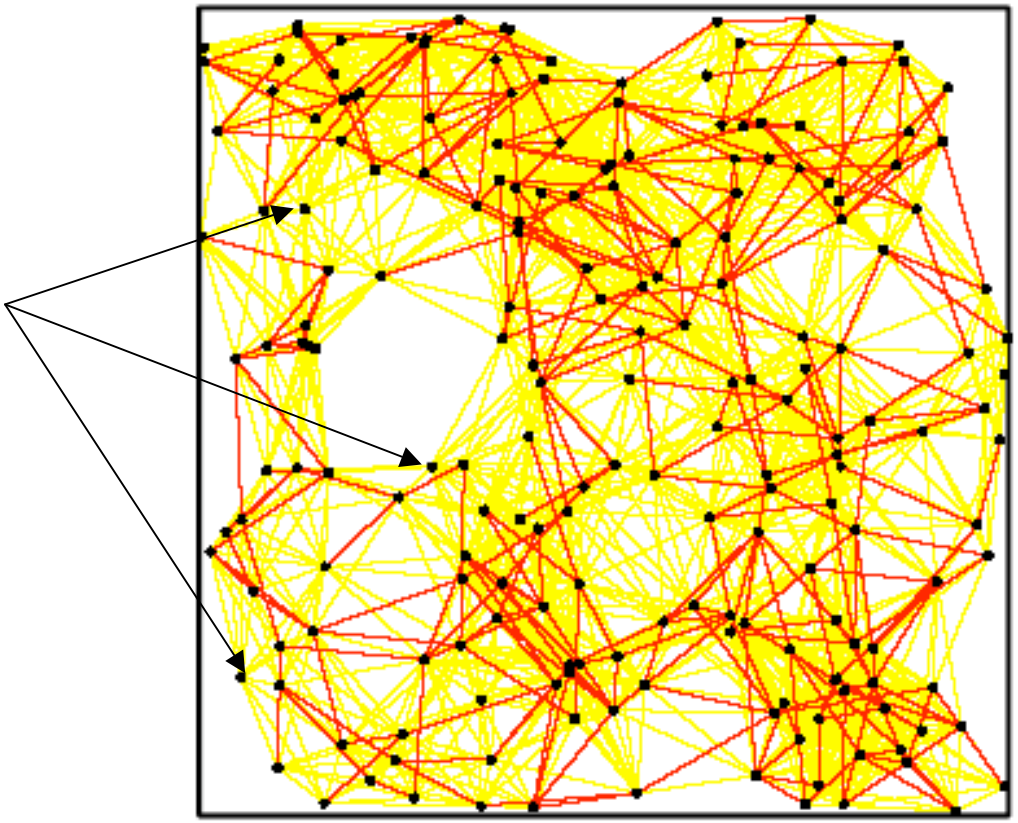
Analytical results confirmed in full

Connectivity



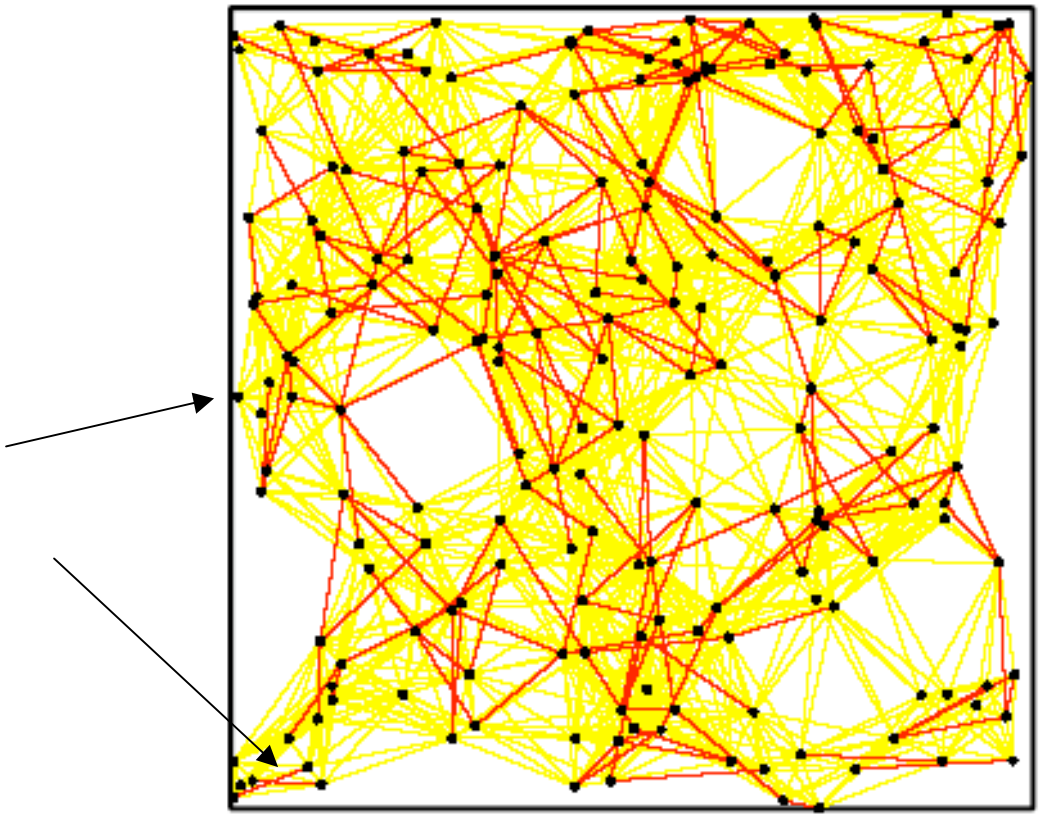
Randomly generated secure sensor network of size 200. Communication range is 0.2, key ring size is 4, and pool size is 50. Lighter lines mean physical visibility, darker lines secure visibility. This graph is connected by using secure links alone.

Giant Component



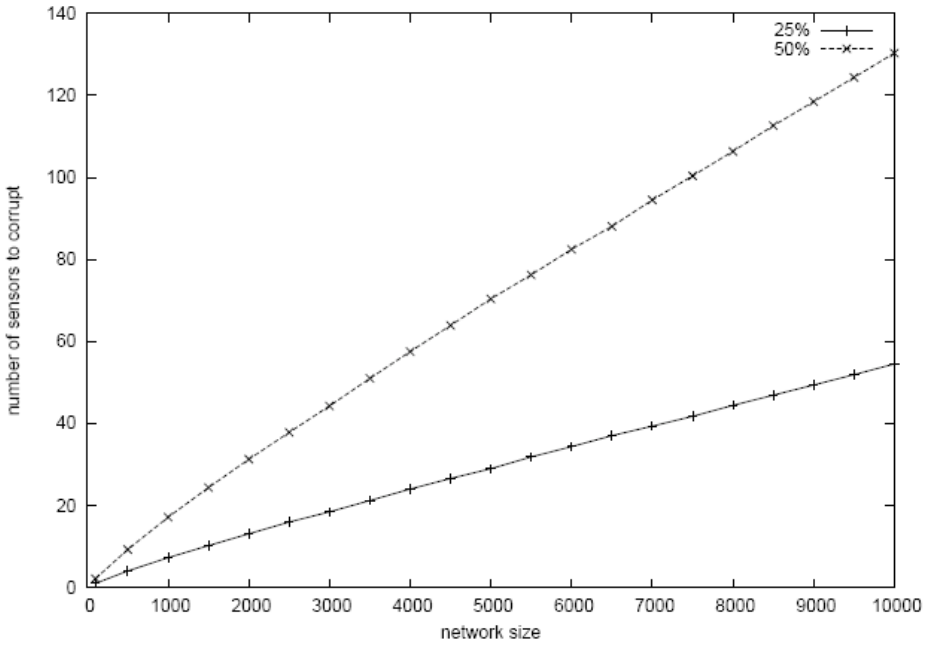
Randomly generated secure sensor network of size 200. Communication range is 0.2, key ring size is 4, and pool size is 100. Lighter lines mean physical visibility, darker lines secure visibility. The network has a few isolated sensors.

Giant Component



Randomly generated secure sensor network of size 200. Communication range is 0.2, key ring size is 4, and pool size is 150. Lighter lines mean physical visibility, darker lines secure visibility. The network has a slightly larger number of isolated sensors and even some very small disconnected components.

Security



Number of sensors that the attacker has to collect to compromise 50% and 25% of the network links.

Observe linear growth, as predicted by theory

Open Problems

Considering security and connectivity properties simultaneously is very fruitful. Future work might consider:

- Different attacks (we want proper definitions for rigorous proofs)
- Other desirable connectivity properties
- Peer-to-peer
- Mobility
- The existence of special, stronger nodes in the network
- Density estimates for good properties to hold

THANKS!