Privacy amplification with social networks

Shishir Nagaraja

Computer Laboratory
JJ Thomson Avenue, Cambridge CB3 0FD, UK
`forename.surname @ cl.cam.ac.uk`

Abstract. There are a number of scenarios where users wishing to communicate, share a weak secret. Often, they are also part of a common social network. Connections (edges) from the social network are represented as shared link keys between participants (vertices). We propose mechanisms that utilise the graph topology of such a network, to increase the entropy of weak pre-shared secrets. Our proposal is based on using random walks to identify a chain of common acquaintances between Alice and Bob, each of which contribute entropy to the final key. Our mechanisms exploit one-wayness and convergence properties of Markovian random walks to, firstly, maximize the set of potential entropy contributors, and second, to resist any contribution from dubious sources such as Sybill sub-networks.

1 Introduction

A secret key agreement protocol is a process by which two or more parties agree on cryptographic keys that can then be used to provide useful communication security properties such as message secrecy and integrity.

One of the basic problems in applying cryptography to real world applications has been the generation of a common secret key between two parties. Informal key agreement protocols such as those based on human chosen passwords can introduce weak keys between the participants for a number of reasons rooted in human psychology and badly designed human-computer interfaces.

In this paper, we show a method of reducing the risk to application security by the compromise of weak keys, by proposing a privacy amplification technique based on graph theory and social network theory.

Previous work has shown that secret key agreement is possible when the participants have access to a noisy common source of random bits. In particular Maurer and Wolf's work [MW03a,MW03b,MW03c] shows that participants knowing correlated pieces of information from such a random source, can generate a common secret key even in the presence of active adversaries over unauthenticated public channels. Satellites broadcasting random strings at low power have been suggested as a means of achieving a common random source. However such centralized infrastructure is undesirable, since a well funded adversary such as a government spy agency could well have a copy of the original random string beamed by the satellite. If the adversary owns the satellite, then key agreement is not possible in the Maurer and Wolf scheme.

More commonly users start out with a weak key such as one based on human readable passwords, that requires amplification before it meets application security needs. In this work, we show how Alice and Bob, can amplify their weak key with a decentralized protocol involving a chain of common acquaintances. Our scheme taps the social network connecting the participants. We show how such parties may generate a stronger key than the initial weak secret. The main contribution of this work is the idea of using Markovian random-walks to obtain a node similarity metric, locating nodes with similar measures to obtain reliable sources of entropy and a decentralized protocol to negotiate the process of entropy contribution.

While most of our analysis surrounds social networks, the application of this work is by no means limited to them. We believe our scheme is fairly practical and should be of use in unstructured decentralized networks wherever low entropy keys are a concern.

2   Background work

Information theoretically secure secret key agreement is a well studied topic, with quite a lot of literature on secret key agreement by public discussion over noisy public channels.

The cryptographic power of a noisy channel was first demonstrated by Wyner [Wyn95] when he showed that two honest parties Alice and Bob can exchange a key over a noisy channel, where the noise between them is less than the Alice-Eve and Bob-Eve channels, where Eve is an eavesdropper.

Maurer [Mau93] then showed that key exchange was possible even if Eve's channel to a binary random source was better (having fewer errors) than that between Alice and Bob, as long as it was independent of the Alice-Bob channel. The value of a binary random source is transmitted to each participant over an independent noisy channel (for instance a satellite beaming down a random stream of numbers with a low signal power). Two parties, say Alice and Bob that receive bits from the same transmission now have an advantage over an eavesdropper Eve, in that they share some common bits of information over Eve. They can then generate a secret key from their received values by public discussion using the cascade protocol of [BS94].

To further minimise the threat posed by Eve, Alice and Bob carry out a step of privacy amplification. Bennet et. al. [BBR88] propose a scheme wherein the n-bit secret key agreed upon by Alice and Bob is mapped to an r-bit string, say by a r*n matrix whose values are chosen at random by one party and send to the other. Both multiply their n-bit key with this matrix to yield an r-bit key about which Eve has negligible information. This is further improved upon in [BBCM95].

We note here that we are not the first to use networks in developing cryptographic primitives. Maurer [BM96] proposed optimal tree based one-time signature schemes. Juels and Peinado [JP00] propose building cryptographic prim-

itives on the hard problem of finding cliques in large random graphs, widely conjectured to be unsolvable in polynomial time.

This paper applies social networks to key-agreement protocols. Social networks have been a fast growing area of interest since the late 90s, due to a combination of factors including the growth of on-line social networks and an explosion of research interest to scientists from a range of disciplines.

Previous work on using social network topologies to enhance system security has been in the context of resisting Sybil attacks [YKGF06] which exploits the mixing properties of social networks to bound the number of Sybil nodes in a decentralized network; Peer-to-peer routing [MGGM04,DLLKA05] exploits social information present in the introduction graph; and trust metrics introduced in [Lev01] use maximum flow in a network to make trust judgements. One-way functions based on graphs have been previously worked on by Bleichenbacher and Maurer, who constructed one-time signature schemes based on directed acyclic graphs [BM96].

3   Threat model

In the information-theoretic secret key agreement with privacy amplification model of Maurer and Wolf [MW03a], Eve the adversary and the communicating parties Alice and Bob, are all assumed to more or less have the same channel noise from the satellite. However, if Eve is the government, and the government runs the satellite (or has access to it) then the assurances of information theoretic secrecy disappear since Eve now has a non-noisy version of the entire random string.

Although the work of Maurer and others performs very impressively under the politically-neutral-satellite threat model, it is completely compromised if Eve is the government.

In our threat model, the adversary is not present during the bootstrap period when activists fearing persecution, Alice and Bob, share a low entropy secret key, such as one based on human memorable passwords. We further assume that the adversary is not able to stop the formation of social networks or completely close off the dissemination of information through them. From a technical perspective our model might appear weaker than the Maurer model, but we claim that it is in fact quite practical and realistic.

4   Motivation

So what can activists like Alice and Bob who wish to communicate securely do when confronted by a well funded adversary like the government? We think using the social network that connects them to their fellow activists could be a way forward. In our scheme, the emphasis is on decentralized ways of key-amplification, that require large scale pre-built infrastructure such as satellites and so on.

The primary motivation for work is the small-world phenomenon - that human participants in a secret key agreement are often part of a common social network connected by a chain of maybe half a dozen others who are pairwise acquainted. This was popularised by early work from Milgram [Mil67] and has been the subject of much research by others.

Social networks range from tiny groups of common friends to large scale on-line networks connecting hundreds of thousands of people. The Internet has seen the growth of a number of on-line social networking web sites such as Orkut [httc], Friendster [htta]
and Livejournal [httb] among others. Graph topologies obtained from such public networks can be used to bootstrap real world implementations of our proposal. Vast amounts of data are also available from social science archives [httd] as well as large scale online social networks.

Consider users on the Internet connected by a social network, who need shared key material to drive various cryptographic operations. Given that a number of application security protocols do give rise to weak keys, is there a neat tool one can offer to users, so that the quality of a weak shared key can be improved? We propose that the key idea behind a viable solution, is to use random walks on the social networks, we detail in the following sections.

5 Privacy amplification in social networks

5.1 Random walks

A random walk on a graph is a stochastic process on a connected graph G of nodes V and edges E, that starts from a randomly selected node v, chooses at random (v, v'), an outgoing edge among the edge set $E(v)$, with a uniform distribution and follows that edge to visit v'. This step is repeated t times, to give a random walk $RW(G, v_{start}, t) = v_{start}, v_1, v_2, ..., v_t$ of length t.

Local operation - Note that a random walk does not rely on the knowing the topology of the entire graph before starting a walk, the node only needs the neighbor list of the currently visited node.

One wayness - It is easy to see the one wayness of this process. Given a node and its edge set, it is easy to pick a target node to visit, but given a visited node v_k at step k, it is much more difficult to trace the starting node. The precise level of difficulty involved in back tracing depends on the topology of the graph and the length of the walk.

If Alice executes a random walk over this network, then, beyond a walk of a certain minimum length, the likelihood that her walk is visiting any particular node tends to a topology specific distribution. This process has the property of a hashing function, in that it is a one-way function with collision resistance as we shall see.

One of the celebrated results from complexity theory is that where one wishes to draw samples from a uniform distribution involving N entities, a random walk that converges on the stationary distribution of the graph will be equally

efficient. The required length of the random walk is an important parameter in determining whether or not the walk converges to the stationary distribution. Hence, given that a random walk starts on a particular node, the final node could be any of the nodes in the network, for the attacker a guess between 1 and N, where N is the number of nodes in the network.

An important concept in random walks on graphs is the stationary distribution, which is the probability distribution of number of visits to the nodes of a graph in a random walk of infinite length.

5.2 Protocols

Alice and Bob start with a pre-shared weak secret based on a human readable password K_{AB}. Both have a purely local view of the network's topology, limited to their respective neighbor lists. The basic idea is that multiple independent walks on well mixing networks overlap substantially. Random walks on such networks can then be used to determine key-amplification paths through the network that have the following desirable properties:

1. *Maximize the anonymity set* - The potential set of entropy contributors is statistically uniformly distributed throughout the network.
2. *Quality entropy contribution* - In order to control the quality of key material generated by the entropy contributors, privacy amplification should only be possible between nodes that are structurally equivalently positioned in the network topology. For instance, if the network in question has three connected sub-components based on a python mailing list, Linux mailing list and a physicists mailing list. if Alice belongs to the python mailing list, then she would like her entropy contributors to be from the same sub-component as her. This prevents dubious entropy contribution from graph components that are only weakly connected to the one she belongs.

Alice initiates a t-step random walk by sending a blob $Hash(N_{Alice}, K_{AB})$ to her neighbor Charlie, Charlie then forwards this to a random neighbor David and so on to Edward, until the walk is completed. Each node on the walk returns a token to Alice, along a suitable return path. Since all inter-node communication is authenticated with inter-node link keys, Alice can only talk to Edward if Alice shares a link key with Edward.

Bob then initiates a random walk of the same length as Alice, and obtains tokens from each node on the walk.

Alice and Bob then sort the list of nodes on their walks, according to an increasing order of integer node-identifiers. The sorted list is divided into a number of blocks according to a mutually agreed block size that equals the number of bits required to represent one node-identifiers. Alice and Bob each calculate the parity of every block and sends this information to the other, using which they can calculate the intersection of their walks. Both, now use the tokens from the intersecting nodes to obtain an identical entropy contribution. Alice and Bob, now share a higher entropy key than K_{AB}. The number of bits of entropy

added, depends on the number of common nodes in the random walk. If they are in the same sub-component of the graph, then there is a higher chance of having common nodes, than if they are in different sub-components.

1. K_{AB} Weak key shared by A and B.
2. d_A Degree of node A.
3. n_A Set of neighbors of node A.
4. N_A Nonce generated by node A.
5. T_A Timestamp generated by A.
6. $blob_A = Hash(N_A, K_{AB})$
7. E_X Random string from node X.
8. $token_{A,X} = \{blob_A, Hash(blob_A, E_X)\}$

Fig. 1: Terminology

$$
\begin{aligned}
0.\ A \longrightarrow B: \quad & \{p_0 = \{A,\quad B,\quad N_A^0, N_A\},\quad Hash(p_0)\}_{K_{AB}} \\
1.\ A \longrightarrow C: \quad & \{p_1 = \{A,\quad C,\quad t,\quad N_A^1,\quad blob_A\},\quad Hash(p_1)\}_{K_{AC}} \\
2.\ C \longrightarrow D: \quad & \{p_2 = \{C,\quad D,\quad t-1,\quad blob_A,\quad N_C\},\quad Hash(p_2)\}_{K_{CD}} \\
3.\ D \longrightarrow E: \quad & \{p_3 = \{D,\quad E,\quad t-2,\quad blob_A,\quad N_D\},\quad Hash(p_3)\}_{K_{DE}} \\
& \dots \\
m.\ X \longrightarrow X': \quad & \{p_t = \{X,\quad X',\quad 1,\quad blob_A,\quad N_X\},\quad Hash(p_t)\}_{K_{XX'}} \\
m+1.\ X' \longrightarrow X: \quad & \{\ p_1' = \{X',\quad X,\quad t,\quad N_X'\},\quad Hash(p_1', token_{A,X'}) \\
& \quad token_{A,X'} = \{blob_A, Hash(blob_A, E_{X'})\}\}_{K_{XX'}} \\
& \dots \\
2m-1.\ C \longrightarrow A: \{\ & \{\ p_t' = \{C,\quad A,\quad N_C',\quad token_{A,X'}, \dots, token_{A,C}\},\quad Hash(p_t')\}_{K_{AC}}\}
\end{aligned}
$$

Fig. 2: Token collection protocol

Alice and Bob, carry out the token collection protocol of Fig 2 to each receive t tokens each. Bob must generate the same blob as Alice ($\forall nodes\ V \in V_1 \dots V_N, token_{A,V} = tokenB, V$ iff A and B share K_{AB}), for which he obtains N_A from Alice in the first message. Next, they engage each-other in the acquaintance-detection protocol of Fig 3 to figure out the nodes they must each contact. Finally, both Alice and Bob request entropy from each of the selected nodes and arrive at a stronger key K_{AB}^{new} as below:

$$Alice: K_{AB}^{new} = Hash(K_{AB}, token_{A,V_1}, \dots, token_{A,V_j})$$

$$Bob: K_{AB}^{new} = Hash(K_{AB}, token_{B,V_1}, \dots, token_{B,V_j})$$

1. $A \longrightarrow B : \{ \ q_1 = \{ \ A, \ B, \ N_A^1, \ (paritybits)P_A^1 \dots \ P_A^t \}, \ Hash(q_1)\}_{K_{AB}}$
2. $B \longrightarrow A : \{ \ q_2 = \{ \ B, \ A, \ N_B^1, P_A^1 \dots \ P_A^t \oplus P_B^1 \dots \ P_B^t \}, \ Hash(q_2)\}_{K_{AB}}$

Fig. 3: Acquaintance detection protocol

5.3 Social salt

The process of deriving a good key from a human readable secret, for use in cryptographic primitives is an important aspect of security engineering, refer [MT79] for an excellent discussion of the topic.

We illustrate a simple method for salt generation in the context of running services requiring simple authentication on low end devices. We note that the vertex sequence resulting from a keyed random walk, offers unique salts for keys that Alice requires. Alice uses a deterministic function whose output is a specific and repeatable sequence of vertices, that can be generated by Alice with K_{AB}, but that is pseudo-random otherwise. The salt is then generated from this sequence.

Instead of a simple random walk $RW(G, v_A, t)$, Alice uses a keyed random walk $RW(G, v_A, t, k_{AB})$ where the sequence of nodes visited $v_A, v_1 .... v_t$ is a function of the G and shared key k_{AB}. At each step t_m Alice uses k_{ab} to decide the next destination v_{m+1} using next $log_2(E(v_m))$ binary bits of k_{AB}. A keyed random walk appears statistically uniformly random to a casual observer of the walk, but, completely deterministic to the someone who knows the starting node and the walk key k_{AB}.

1. Alice and Bob share a secret key k_{AB}.
2. Alice selects a starting node, which could be Alice's own node v_A.
3. Using $log_2(out - degree_A)$ bits of k, she selects one of the outgoing edges (A, X_1).
4. She then queries X_1 for his neighbor list, or uses the locally cached topology, and repeats the previous step, until all the entropy of k_{AB} has been used up in constructing the walk $RW(G, k_{AB}, v_A, t) = v_A, v_1, v_2, ..., v_{t-1}, v_t$ of length t. Where $v_1, v_2, ..., v_{t-1}$ is the set of intermediate nodes and $v_{(A,k_{AB},t)}$ is the destination of the walk starting from node A with key k_{AB}.
5. Alice now performs $RW(G, k_{AB}, B, t)$ to obtain $v_{(B,k_{AB},t)}$.
6. Alice then generates a new key

$$K_{AB}^{new} = Hash(v_{(A,k_{AB},t)}, v_{(B,k_{AB},t)}, k_{AB})$$

5.4 Networks

In a graph $G(V, E)$, where V is the set of vertices or nodes of a social network and E is the set of edges or links connecting the nodes. The distribution of

probabilities of a randomly selected node having exactly k links $Pr[degree(x) = k]$, over all the nodes is known as the *degree distribution of a network*.

Early work by Erdös and Renyi modelled networks as random graphs [ER59,Bol01]; this is mathematically interesting but does not model most real-world networks accurately. However for completeness we shall include this model in our study. In real networks, path lengths are generally shorter; it is well known that any two people are linked by a chain of maybe half a dozen others who are pairwise acquainted – known as the 'small-world' phenomenon. This idea was popularised by Milgram in the 60s [Mil67]. Then in 1998 Watts and Strogatz produced the alpha model. Alpha is a parameter that expresses the tendency of nodes to introduce their neighbors to each other; with $\alpha = 0$, each node is connected to its neighbors' neighbors, so the network is a set of disconnected cliques, while with $\alpha = \infty$, we have a random graph. They discovered that, for critical values of α, a small-world network resulted. The alpha model is rather complex to analyse, so they next introduced the beta network: this is constructed by arranging nodes in a ring, each node being connected to its r neighbors on either side, then replacing existing links with random links according to a parameter β; for $\beta = 0$ no links are replaced, and for $\beta = 1$ all links have been replaced, so that the network has again become a random graph [WS98]. The effect is to provide a mix of local and long-distance links that models observed phenomena in social and other networks. We use a version of this model to explain the significance of our work.

How do networks with short path lengths come about in the real world? The simplest explanation involves preferential attachment. Barabási and Albert showed in 1999 how, if new nodes in a network prefer to attach to nodes that already have many edges, this leads to a power-law degree distribution which in turn gives rise to a *scale-free* network [BAJ99], which turns out to be a more common type of network than the alpha or beta types. In a social network, for example, people who already have many friends are useful to know, so their friendship is particularly sought by newcomers. In friendship terms, the rich get richer. The Barabási-Albert(BA) model suffers from one major disadvantage. While real world social networks have high clustering coefficients (the probability that the friend of your friend is your friend or the ratio of observed triads in the network over all possible triads), the BA model does not exhibit the high clustering one finds in real world networks. Additionally, while a number of real world social networks are scale-free, many have turned out not to be so, lying anywhere between poisson random networks and completely structured k-regular networks.

Hence, we also include in our study a small world network model. Networks characterised by small shortest paths and high local clustering coefficients. We use the small world model introduced by Watts-Strogatz [WS98] and subsequently modified by Klienberg [Kle00] that encapsulates rich local links with a few long range connections. In the rest of the text we refer to this as the Klienberg-Watts-Strogatz (KWS) model.

6 Mixing properties of social networks

In earlier sections we have shown how random walks can be used for key-amplification in distributed decentralized networks. We now carry out some analysis to figure out the length of the walk to obtain the desired property of maximizing the number of potential entropy contributors.

In order to get a reasonable idea about the mixing properties of different types of networks we consider a number of network models and comment on their mixing properties. We derive analytical results for scale-free networks and obtain simulation based results for other networks. We then comment on the recommended walk lengths for each case. In each network model, we ran generated 100 instances of the network for the specified parameters and averaged the results.

The number of steps to convergence, of a random walk on a network is the number of steps in the walk, so that the probability of visiting a randomly selected node at the end of the walk, is (nearly) equal to that at the end of a walk of infinite length.

The quality of privacy amplification from a random walk, then depends on both the number of possible final walk destinations and the probability distribution of any node in the network being the destination node, $Pr[v_{(X,k_{CY},t)}) = v_i]$.

To measure the convergence rates of these networks we use an information theoretic based metric of entropy. The entropy of a probability distribution is defined as follows:

$$\mathcal{E}[v_i] = -\sum_i Pr[v_i]log_2 Pr[v_i] \tag{1}$$

We use a normalized form of en 1, normalized entropy0 of the network. As the number of steps in the walk increases the mean entropy catches up to the maximum value of 1. The rate at which this happens can be useful to characterise a network.

6.1 Erdös-Rényi model of random networks

In the Erdös-Rényi (ER)model [ER59], we start from N vertices without any edges. Subsequently, edges connecting two randomly chosen vertices are added as the result of a Bernoulli trial, with a parameter p. It generates random networks with no particular structural bias. The average degree $< k >= 2L/N$ where L is the total number of edges, can also be used as a control parameter. ER model networks have a logarithmically increasing l, a normal degree distribution, and a clustering coefficient close to zero.

We used N=5000 nodes, $< k >$=14, and p=0.0014. The analysis of mixing rates is at the end of the next subsection 6.2.

Although no social networks look like ER networks, we included this model to provide a baseline comparison with other networks.

6.2 Scale-free networks with linear preferential attachment

A variable X is said to follow a heavy tail distribution if $Pr[X > x] \sim x^{-k} L(x)$ where $k \in \Re^+$ and $L(x)$ is a slowly varying function so that $\lim_{x\to\infty} \frac{L(tx)}{L(x)} \to 1$. A power-law distribution is simply a variation of the above where one studies $Pr[X = x] \sim x^{-(k+1)} = x^{-\alpha}$. The degree of a node is the number of links it has to other nodes in the network. If the degree distribution of a network follows a power-law distribution it is known as a scale-free network. The power-law in the degree or link distribution reflects the presence of central individuals who interact with many others on a continual basis and play a key role in relaying information.

We denote a scale-free network generated by preferential attachment, by $G_{m,N}(V, E)$ where m is the number of initial nodes created at time=t_0 and N is the total number of nodes in the network. At every time step $t_i, i \geq 0$, one node is added to the network. For every node v added, we create m edges from the v to existing nodes in the network according to the following linear preferential attachment function:

$$Pr[(v, i)] = k_i / \sum_j k_j$$

where k_i is the degree of node i. We continue until $|V| = N$.

$$\Delta(t) = max_i \frac{|q_i^t - \pi_i|}{\pi_i} \tag{2}$$

In order to convince ourselves that scale-free networks mix well, we first prove that the mixing rate is independent of N. We then show empirically that t in eqn. 2 is reasonably small to support the proposed protocol in fig 2.

There is an intimate relationship between the rate of convergence and a certain structural property called the *conductance* of the underlying graph. Consider a randomly chosen subgraph S of $G(V, E)$. Suppose a random walk on the graph visits node i, $i \in S$. What is the probability that the walk exits S in a single hop. If conductance is small, then a walk would tend to "get stuck" in S, whereas if conductance is large it easily "flows" out of S.

Formally, for $S \subset G$, the *volume* of S is $vol_G(S) = \sum_{u\in S} d_G(u)$, where $d_G(u)$ is the degree of node u. The *cutset* of S, $C_G(S, \overline{S})$, is the multi-set of edges with one endpoint in S and the other endpoint in $\overline{S}$. The textbook definition of conductance Φ_G of the graph G is the following:

$$\Phi_G = \min_{S\subset V, vol_G(S) \leq vol_G(V)/2} \frac{|C_G(S, \overline{S})|}{vol_G(S)} \tag{3}$$

From [Sin93] we have the following bound for λ_2 in terms of *conductance*

$$1 - 2\Phi \leq \lambda_2 \leq 1 - \Phi^2/2 \tag{4}$$

[MPS03] prove that the conductance of a scale-free network is a *constant*. Specifically, $\forall m \geq 2$ *and* $c < 2(d-1) - 1$, $\exists \alpha = \alpha(d, c)$ such that

$$\Phi = \frac{\alpha}{m + \alpha} \qquad (5)$$

From eqn. 4 and eqn. 5 we conclude that the 2nd eigenvalue of the transition matrix of a scale-free network is a *constant*, and is independent of N.

The figure 4 shows how the entropy of the chain increases with increasing length of the random walk. One can conclude that social networks that can be modelled by a BA-scale-free network can be potentially successful far as walk lengths are concerned with a convergence in $O(log\ N)$ steps of the random walk. However complexity theoretic bounds are not very useful to the system designer, hence we carried out simulations to get a better idea of the required walk length.

Specifically, in the simulations above, for a network of size 5000 nodes and average degree $< d >= 14$ in scale-free and ER topologies, we observe that the scale-free topology reaches the stationary distribution in 6 hops while ER is much slower at 7 hops.
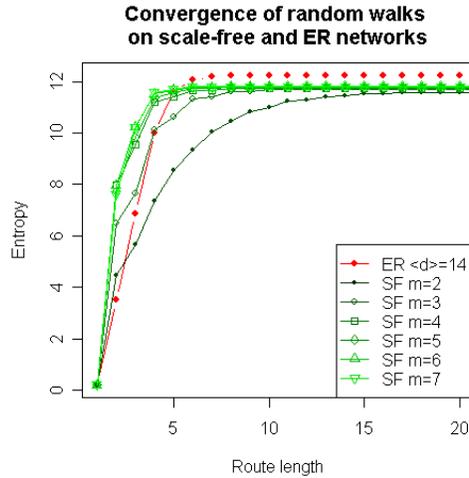

Fig. 4: normalised entropy vs walk length

6.3   Scale-free random graph topology

Since ER graphs do not capture the scale-free character of real networks, we use an input degree distribution vector that is restricted to a power-law but is random in all other aspects. We use the threshold model of Aiello et. al. [ACL00]

with a slope of 2.5 and a fixed number of nodes $N = 5000$ with average degrees of 2,3,4,5 and 6 in separate instances of the graph.

We then extracted the largest connected component of the graph and used it in our anonymity analysis as per the framework used in previous sections.
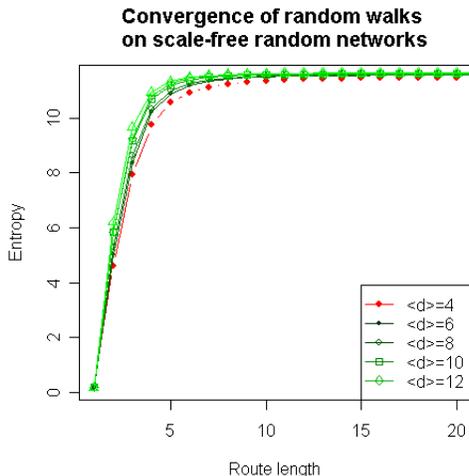
Convergence of random walks on scale-free random networks

Fig. 5: normalised entropy vs walk length

Simulation results are shown in Fig 5. Scale-free random topology takes approximately 8 steps to reach stationary distribution. So, scale-free random networks approximately take one extra step compared to BA scale-free networks. We think, this is attributable to the relatively much higher clustering coefficient of scale-free random graphs.

6.4 Klienberg Watts-Strogatz (KWS) small world topology

The Klienberg WS graph topology models a small world network that is rich in local connections, with a few long range connections. The network generation starts from a N by N lattice each point representing an individual in a social network. The lattice distance $d((i, j), (k, l)) = |k - i| + |l - j|$. For a parameter p, every node u has a directed link to every other node v within $d(u, v) \leq p$. For parameters q and r, we construct q long range directed links from u to a node v with a probability distribution $[Pr(u, v)] = \frac{(d(u,v))^{(-r)}}{\sum_v (d(u,v))^{(-r)}}$.

Low r values means long-range connections, whereas higher values lead to preferential connections in the vicinity of u.

Fig 6 shows the mixing rates in a small world network as modelled by the Klienberg WS topology. In our simulations we used $N = 5041$ nodes, with $p = 3$,
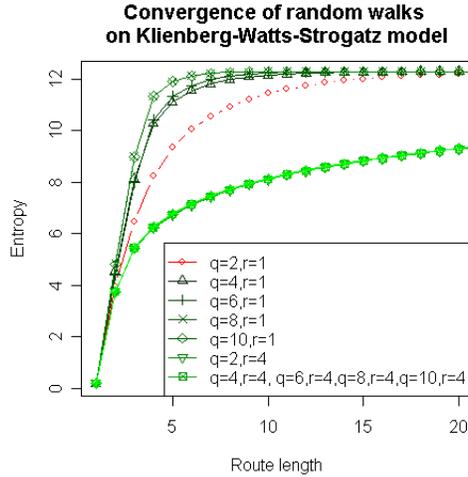
**Convergence of random walks
on Klienberg-Watts-Strogatz model**

Legend:
- q=2,r=1
- q=4,r=1
- q=6,r=1
- q=8,r=1
- q=10,r=1
- q=2,r=4
- q=4,r=4, q=6,r=4,q=8,r=4,q=10,r=4

Fig. 6: normalised entropy vs walk length

q values of 2,4,6,8,10, and r values of 1,5. The graphs show that with higher values of r the mixing rate is poor and so is the normalised entropy of the stationary distribution. For $r = 5$, normalized entropy< 0.8 even after 20 rounds. However, for $r = 1$, it reaches maximal entropy within 6-7 steps.

This means, that small world networks that have very high clustering, and a few long range connections are well suited for our application. However, those with few to no long range connections might be troublesome candidates.

7 A note on theoretical topologies

While theoretical networks such as expander graphs, de-Bruijn graphs and Ramanujan graphs will technically work better in our scheme thanks to better mixing properties, we have no interest in them, since real world social networks hardly resemble the regularity such models exhibit. In the context of key amplification, the network's edges represent trust relationships between nodes, hence it is not possible to consider structured graphs.

There are other reasons why structured graphs may be undesirable, we explain this as follows. Structured overlay networks require higher amounts of resources to maintain the same number of edges in the network - since the social incentives that keep edges in place in a social network are absent in a structured overlay network. For instance, willingness to communicate and pass on information tends to be higher between friends than between strangers. Additionally, structured network topologies assume degree homogeneity (all nodes have the same number of links), whereas the diversity in real world node capabilities in peer-to-peer networks and router networks suggests otherwise. The AS

graph constructed from the BGP routing tables are a case in point, see [FFF99] for 1997-98 and [MKF$^+$05] for a more recent version. Gnutella [KM02] and Freenet [CSWH00] both popular peer-to-peer systems are found to have significantly skewed node degrees according to the measurements of [RFI02].

8 Conclusions

The problem of boot strapping strong keys among users in online communication has been a basic problem in cryptography for a number of years now. A number of solutions in the past have relied on centralized infrastructure. We suggest a different approach of bootstrapping key establishment from a pre-existing trust network, such as a social network that has been shown by social scientists to connect any two humans by a shortest path of six or so hops.

We show how, with a merely local view of (social) network topology, Alice and Bob conduct independent random walks to identify a set common of acquaintances dispersed throughout the network. Each member of this set then contributes entropy to the weak secret Alice and Bob share. We then analyze a number of network models for their mixing properties, and conclude that a walk length of seven hops is sufficient in all networks, to maximize the set of potential contributors.

Our protocols also provide some level of assurance to Alice that the other party Bob, is not a member of a Sybill group. This is achieved by using the topological properties of the underlying network as shown by the work of Yu et.al. [YKGF06]. While the focus in this paper has been the presentation and evaluation of our scheme on social networks, it should be readily extendable to most decentralized computing scenarios such as sensor and adhoc networks.

We have shown that such a scheme is realistic, given that in all the model and practical networks we have considered in this study, the number of steps required to reach the stationary distribution is $O(log(N))$.

Future work in the area involves the use of position similarity metrics to arrive at other meaningful ways of choosing protocol participants. In addition we aim to calculate precisely the length of the walk execute by Alice and Bob each, to obtain k common partners in the walk.

9 Acknowledgements

References

[ACL00]   William Aiello, Fan Chung, and Linyuan Lu. A random graph model for massive graphs. In *STOC '00: Proceedings of the thirty-second annual*

     ACM symposium on Theory of computing, pages 171–180, New York, NY, USA, 2000. ACM Press.

[BAJ99] Albert-Laszlo Barabasi, Reka Albert, and Hawoong Jeong. Mean-field theory for scale-free random networks. *Physica A*, 272:173–187, 1999.

[BBCM95] Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Ueli Maurer. Generalized privacy amplification. *IEEE Transaction on Information Theory*, 41(6):1915–1923, Nov 1995.

[BBR88] Charles H. Bennett, Gilles Brassard, and Jean-Marc Robert. Privacy amplification by public discussion. *SIAM J. Comput.*, 17(2):210–229, 1988.

[BM96]   Daniel Bleichenbacher and Ueli Maurer. Optimal tree-based one-time digital signature schemes. In C. Puech and R. Reischuk, editors, *Proc. 13th Symposium on Theoretical Aspects of Computer Science (STACS'96)*, volume 1046 of *Lecture Notes in Computer Science*, pages 363–374. Springer-Verlag, February 1996.

[Bol01]   B. Bollobas. *Random Graphs*. Cambridge University Press, 2001.

[BS94]   Gilles Brassard and Louis Salvail. Secret-key reconciliation by public discussion. In *EUROCRYPT '93: Workshop on the theory and application of cryptographic techniques on Advances in cryptology*, pages 410–423, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc.

[CSWH00] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore Hong. Freenet: A distributed anonymous information storage and retrieval system. In *Proceedings of Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability*, pages 46–66, July 2000.

[DLLKA05] George Danezis, Chris Lesniewski-Laas, M. Frans Kaashoek, and Ross Anderson. Sybil-resistant dht routing. In *Proceedings of the 10th European Symposium On Research In Computer Security*, Milan, Italy, September 2005.

[ER59]   P. Erdos and A. Rnyi. On random graphs. *Publicationes Mathemticae (Debrecen)*, 6:290–297, 1959.

[FFF99]   Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On power-law relationships of the internet topology. In *SIGCOMM '99: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, pages 251–262, New York, NY, USA, 1999. ACM Press.

[htta]   http://. www.friendster.com.

[httb]   http://. www.livejournal.com.

[httc]   http://. www.orkut.com.

[httd]   http://www.esds.ac.uk. esds - economic and social data service.

[JP00]   Ari Juels and Marcus Peinado. Hiding cliques for cryptographic security. *Designs, Codes and Cryptography*, 20(3):269–280, July 2000.

[Kle00]   Jon Kleinberg. The Small-World Phenomenon: An Algorithmic Perspective. In *Proceedings of the 32nd ACM Symposium on Theory of Computing*, 2000.

[KM02]   T. Klingberg and R. Manfredi. "gnutella 0.6", June 2002.

[Lev01]   R. Levien. Attack-resistant trust metrics, 2001.

[Mau93] Ueli Maurer. Secret key agreement by public discussion. *IEEE Transaction on Information Theory*, 39(3):733–742, May 1993.

[MGGM04] Sergio Marti, Prasanna Ganesan, and Hector Garcia-Molina. Dht routing using social links. In *IPTPS*, pages 100–111, 2004.

[Mil67]   Stanley Milgram. The small world problem. *Psychology Today*, 2:60–67, 1967.

[MKF+05] Priya Mahadevan, Dmitri Krioukov, Marina Fomenkov, Bradley Huffaker, Xenofontas Dimitropoulos, K C Claffy, and Amin Vahdat. Lessons from three views of the internet topology. Technical report, Cooperative Association for Internet Data Analysis (CAIDA), 2005.

[MPS03] Milena Mihail, Christos Papadimitriou, and Amin Saberi. On certain connectivity properties of the internet topology. In *FOCS '03: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, page 28, Washington, DC, USA, 2003. IEEE Computer Society.

[MT79] Robert Morris and Ken Thompson. Password security: A case history. *CACM*, 22(11):594–597, 1979.

[MW03a] Ueli Maurer and Stefan Wolf. Secret key agreement over a non-authenticated channel — part i: Definitions and bounds. *IEEE Transactions on Information Theory*, 49(4):822–831, April 2003.

[MW03b] Ueli Maurer and Stefan Wolf. Secret key agreement over a non-authenticated channel — part ii: The simulatability condition. *IEEE Transactions on Information Theory*, 49(4):832–838, April 2003.

[MW03c] Ueli Maurer and Stefan Wolf. Secret key agreement over a non-authenticated channel — part iii: Privacy amplification. *IEEE Transactions on Information Theory*, 49(4):839–851, April 2003.

[RFI02] Matei Ripeanu, Ian Foster, and Adriana Iamnitchi. "mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design". *IEEE Internet Computing Journal*, 6(1), August 2002.

[Sin93] Alistair Sinclair. *Algorithms for random generation and counting: a Markov chain approach*. Birkhauser Verlag, Basel, Switzerland, Switzerland, 1993.

[WS98] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998.

[Wyn95] Aaron D. Wyner. The wire-tap channel. *Bell Systems Technical Journal*, 41(54):1355–1387, Nov 1995.

[YKGF06] Haifeng Yu, Michael Kaminsky, Phillip B. Gibbons, and Abraham Flaxman. Sybilguard: defending against sybil attacks via social networks. In *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 267–278, New York, NY, USA, 2006. ACM Press.