KIPS Tr. Comp. and Comm. Sys.
Vol.2, No.12 pp.533~540 pISSN: 2287-5891

http://dx.doi.org/10.3745/KTCCS.2013.2.12.533

Extraction of System-Wide Sybil-Resistant Trust Value embedded in Online Social Network Graph

Kyungbaek Kim†

ABSTRACT

Anonymity is the one of main reasons for substantial improvement of Internet. It encourages various users to express their opinion freely and helps Internet based distributed systems vitalize. But, anonymity can cause unexpected threats because personal information of an online user is hidden. Especially, distributed systems are threatened by Sybil attack, where one malicious user creates and manages multiple fake online identities. To prevent Sybil attack, the traditional solutions include increasing the complexity of identity generation and mapping online identities to real-world identities. But, even though the high complexity of identity generation increases the generation cost of Sybil identities, eventually they are generated and there is no further way to suppress their activity. Also, the mapping between online identities and real identities may cause high possibility of losing anonymity. Recently, some methods using online social network to prevent Sybil attack are researched. In this paper, a new method is proposed for extracting a user's system-wide Sybil-resistant trust value by using the properties embedded in online social network graphs. The proposed method can be categorized into 3 types based on sampling and decision strategies. By using graphs sampled from Facebook, the performance of the 3 types of the proposed method is evaluated. Moreover, the impact of Sybil attack on nodes with different characteristics is evaluated in order to understand the behavior of Sybil attack.

Keywords : Sybil Attack, Sybil-Resistant Trust Value, Fast Mixing Graph, Online Social Network

온라인 소셜 네트워크 그래프에 내포된 시스템-차원 시빌-저항 신뢰도 추출

김 경 백†

요 약

인터넷의 발달의 주요 요인 중 하나인 익명성은 다수 사용자들의 자유로운 개인 의사 표현을 도와 다양한 인터넷 기반 분산시스템을 활성화 하는데 있어 큰 도움이 되어 왔다. 하지만, 익명성은 개인의 정보가 외부로 알려지지 않는 다는 점 때문에 악용될 소지도 다분하다. 특히 분산시스템은 한 명의 악의적인 사용자가 다수의 가짜 신분을 생성하고 조정하는 시빌 어택(Sybil Attack)에 매우 취약하게 된다. 시빌 어택을 막기 위해서 분산시스템 상에서 신분 생성 작업의 복잡도를 높이는 방식이나 시스템상의 신분과 현실상의 신분의 연결 고리를 만드는 방법을 생각 할 수 있다. 하지만 복잡도를 높이는 방식은 가짜 신분이 만들어지는 시간을 늘리는 효과만 있을 뿐, 일단 가짜 신분이 만들어진 이후에 대한 대응법이 부족하다. 또한, 현실상의 신분과의 연결을 사용할 경우 온라인 사용자의 익명성이 훼손당할 우려가 있다. 최근 온라인 소셜 네트워크의 대중화와 함께 소셜 네트워크 그래프 정보를 사용해 시빌 어택에 대응하기 위한 기법들이 연구되고 있다. 이 논문에서는 온라인 소셜 네트워크 그래프에 내포된 특성을 이용해 임의의 사용자에 대한 시스템 차원 시빌-저항 신뢰도(System-wide Sybil-resistant trust value) 추출 방법을 제안한다. 제안하는 기법은 온라인 소셜 네트워크 전체 그래프를 이해 할 수 있는 서비스 제공자들을 위한 방법으로, 샘플링 및 판단방법에 따라 3가지 종류의 세부 기법들을 제안한다. Facebook에서 추출한 온라인 소셜 네트워크 샘플 그래프를 이용하여 제안된 기법들의 성능을 분석 및 비교한다. 또한 시빌 어택의 특성을 이해하기 위해 서로 다른 노드 특성을 가지는 노드들이 시빌 어택에 의해 받는 영향을 분석한다.

키워드 : 시빌 어택, 시스템-차원 시빌-저항 신뢰도, 패스트 믹싱 그래프, 온라인 소셜 네트워크

※ 본 연구는 미래창조과학부 및 정보통신산업진흥원의 대학IT연구센터육성지원사업의 연구결과로 수행되었음(NIPA-2013-H0301-13-3005).
※ 이 논문은 SMA 2012에서 발표된 논문[14]을 확장한 논문임.

† 정 회 원 : 전남대학교 전자컴퓨터공학부 조교수
논문접수 : 2013년 10월 7일
심사완료 : 2013년 11월 25일
* Corresponding Author : Kyungbaek Kim(kyungbaekkim@chonnam.ac.kr)

1. 서 론

오늘날 인터넷 상의 분산 시스템은 빠르게 대중화 되어 가고 있다. 인터넷 상의 분산 시스템은 다양한 특성을 가진 구성요소로 이루어져 있을 뿐 만 아니라 다양한 사용자들에 의해서 운용된다. 분산 시스템의 개방적 특성은 임의의 사용자들이 익명성을 가지고 시스템에 정보를 제공하도록 유도하고, 사용자간의 협업을 촉진 시킨다. 하지만, 개방적 특성은 제공되는 정보의 신뢰성을 판단하기 힘들게 한다. 이러한 문제를 해결하기 위해 사용자들의 과거의 행동 양식에 기반을 둔 신뢰도 추출 방식을 이용하는 Reputation 시스템들이 연구 되었지만[1][2], 이들 시스템들에서는 과거의 기록이 없는 새로운 사용자들에 대한 신뢰도 측정이 힘들게 된다. 이와 같은 이유로 개방형 분산 시스템은 시빌 어택(Sybil Attack)에 취약하게 된다[3]. 시빌 어택은 한명의 사용자가 다수의 가상/가짜 사용자들을 생성하여 분산 시스템에서 불공정한 이득을 취하거나 시스템을 마비시키는 공격이다. 시빌 어택에서 사용되는 가상의 사용자들은 시빌 사용자(Sybil User)라고 불리고, 이 시빌 사용자들은 비교적 손쉽게 가공/폐기가 가능하다. 이와 같은 이유로 시빌 어택을 일반적인 과거 행동 방식 기반의 Reputation 시스템에서 방어한다는 것은 쉽지 않다.

시빌 어택에 대항하기 위한 전통적인 방법으로 사용자 생성의 복잡도를 높이는 CAPTCHA[4]를 생각 할 수 있다. 즉 악의적인 사용자가 가상의 시빌 사용자를 생성할 때 필요한 시간적/물리적 비용을 증가시켜, 시빌 사용자의 생성의 빈도를 줄인다는 것이다. 하지만 이 방식의 문제는 비록 생성 비용이 더 들어가긴 하지만, 결국 시빌 사용자는 생성 되고 생성된 시빌 사용자들의 공격을 억제하기 위한 기능이 없다는 것이다. 또 다른 방법으로는 주민 번호나 신용카드 번호와 같은 실제 세상의 사용자 정보와 온라인 상의 사용자 정보를 연결한 맵핑을 사용하는 방식이 있다. 이 방식은 가상의 사용자 생성 억제에 있어 아주 효과적이긴 하지만, 사용자 정보 유출과 같은 새로운 문제를 발생 시킬 수 있다.

최근 이러한 문제 해결을 위해 온라인 소셜 네트워크(OSN)그래프를 기반으로 하는 시빌 사용자 확인 기법들이 제안되었다[5][6][7]. OSN 그래프는 실제 사용자들의 관계를 나타내게 되고, Sybil 사용자들은 이들을 생성한 사용자를 제외한 다른 실제 사용자들과 OSN 그래프 상에서 잘 연결되지 않는다는 특징이 있다[5][6]. 이와 같은 OSN 그래프의 특징을 이용하여 임의의 사용자를 실제 사용자로 받아들일지 아니면 시빌 사용자로 여길지를 판단하게 된다. 이때, 사용자의 과거의 행동 기록을 통해 판단하는 것이 아니라, 각 사용자가 시스템에 참여하는 과정을 통해 판단된다. 따라서, 시빌 사용자들이 새롭게 생성되더라도 그 생성 과정에서 발생하는 OSN 그래프 상의 관계를 통해 임의의 사용자는 이 새로운 사용자들이 시빌 사용자임을 확인할 수 있게 된다.

이와 같은 기술들은 주로 분산시스템 환경에서 임의의 하나의 노드가 다른 노드의 시빌 여부를 확인하는데 사용된다. 하지만, 온라인 소셜 네트워크 그래프 기반의 시빌 사용자 확인 기법들은 네트워크 및 노드의 특성에 따라 영향을 받기 때문에, 개별 노드가 임의의 다른 노드의 시빌 여부를 판단하는데 있어 그 편차가 크게 발생 할 수 있다[12][13]. 이러한 큰 편차는 시빌 사용자들이 특정 사용자를 공격하는 것과 같은 새로운 문제를 야기 할 수 있다.

이 논문에서는 개인 사용자가 아닌 전체 시스템에서 인지할 수 있는 시스템-차원 시빌-저항 신뢰도를 온라인 소셜 네트워크의 특성을 이용해 추출하기 위한 방법들을 제안한다. 제안된 추출 기법은 온라인 소셜 네트워크 상의 정상적인 사용자들은 복잡한 연결 관계를 가지며 그래프를 구성하는 반면, 시빌 사용자들은 일반적인 사용자들과의 연결 관계가 제한적인 점을 고려한 랜덤 워크(Random Walk) 기반의 샘플링을 이용한다. 샘플링 방식에 따라, RRI(Random Route Intersection), RRTI(Random Route Tail Intersection) 그리고 RWTI(Random Walk Tail Intersection), 총 3가지의 서로 다른 추출 기법 타입들을 제안한다. 또한, 개인 사용자 측면이 아닌 시스템-차원의 신뢰도를 추출하기 위해, 단일 판별자를 사용하지 않고 온라인 소셜 네트워크 그래프로부터 다수의 판별자를 선택하여 사용한다.

제안되는 기법들의 성능 평가를 위해 Facebook에서 추출한 샘플 그래프를 사용하여 다양한 실험을 수행하였고, 이를 통해 각 기법들에 대한 특징을 분석하였다. 또한, 추출된 신뢰도의 응용방식 이해를 위해 P2P 파일 공유 시스템에 시스템-차원 시빌-저항 신뢰도값을 적용하였을 때의 시스템의 성능을 측정하고 그 결과를 분석하였다.

이 논문의 구성은 다음과 같다. 2장에서는 온라인 소셜 네트워크에 내포된 시빌-저항 신뢰도 특성을 이해하기 위한 배경 및 가정을 기술하고 이와 관련된 연구에 대한 소개한다. 3장에서는 시스템-차원 시빌-저항 신뢰도를 추출하기위한 샘플링 및 판별 기법들에 대해 설명한다. 제안된 기법들에 대한 성능 검증 및 분산시스템에 적용된 결과 분석 등을 4장에서 기술하고, 5장에서 이 논문의 결론을 말한다.

2. 배경 및 관련 연구

2.1 배경 및 가정

임의의 그래프 G=(V,E) 가 있다고 할 때, 이 그래프의 노드(node)는 $V=\{v_1,v_2,...,v_N\}, |V|=N$ 로 표현하고 에지(edge)는 $E=\{e_{ij}|v_i \leftrightarrow v_j\}, |E|=M$ 이라고 하고, 임의의 온라인 소셜 네트워크를 이 그래프 모델에 매치시킬 수 있다. 즉 각 노드는 소셜 네트워크의 각 사용자와 연결 지어 생각하고, 각 에지는 소셜 네트워크 사용자들 간의 신뢰 관계를 나타낸다. 즉 임의의 두 사용자가 서로를 신뢰한다고 할 때 두 사용자 사이에 Undirected 에지가 존재한다고 가정한다. 일반적으로 소셜 네트워크는 아주 복잡하게 연결되어 있고, 임의의 소셜 네트워크 그래프는 하나의 strongly connected component라고 가정한다.

소셜 네트워크 그래프에서 임의의 정직한 사용자(honest user)는 일반적으로 하나의 노드로 표현된다. 반면, 악의적

인 사용자(malicious user)는 다수의 시빌 사용자를 생성하고 이 다수의 시빌 사용자들은 소셜 그래프의 다수의 노드들로 표현된다. 이와 같은 노드 표현을 기반으로 소셜 네트워크 그래프는 Honest Region과 Sybil Region의 두 개의 구역으로 표현된다. Honest Region은 정직한 사용자들로 구성된 네트워크 구역으로 하나의 strongly connected component이다. 즉 이 Honest Region내에서는 임의의 두 정직한 사용자를 연결하는 에지의 모음이 있다는 것이다. Sybil Region또한 하나의 strongly connected component로서, 시빌 사용자들로 구성되어 있다. 이러한 두 구역들이 서로 연결되어 하나의 strongly connected component를 구성하는데, 이 두 구역을 연결하는 에지를 attack edge라고 한다. 이 attack edge는 정직한 사용자가 시빌 사용자에게 속아서 잘못된 신뢰 관계를 생성한 경우에 만들어진다. 따라서 만약 정직한 사용자들이 충분히 주의를 기울여 신뢰 관계를 관리한다면 이 attack edge의 수는 아주 적게 된다.

이때, 주목할 점은 Honest Region의 임의의 정직한 사용자와 Sybil Region의 임의의 시빌 사용자를 연결하는 에지의 모음에는 반드시 attack edge가 포함되어야 한다는 점이다. 따라서 일반적인 환경에서 attack edge가 많이 생성될 수 없다는 점을 고려하면, 임의의 정직한 사용자에서 시작한 랜덤 워크가 attack edge를 지나 시빌 사용자에게 도달할 확률이 아주 낮다고 생각 할 수 있다. 이를 확장 시켜 생각한다면, 임의의 정직한 사용자가 랜덤 워크 방식을 이용한 소셜 네트워크 정보의 샘플들을 수집할 경우 시빌 사용자가 수집하게 되는 샘플과 그 결과가 많이 다를 것으로 예상 할 수 있다.

2.2 관련 연구

최근 연구된 온라인 소셜 네트워크 기반의 시빌 사용자 확인 기법들은 모두 위와 같은 가정을 기반으로 연구되어 왔다[5][6][7]. SybilGuard[5]와 SybilLimit[6]은 Random Route라는 변형된 Random Walk개념을 사용한 샘플링 비교 방식을 통해, 임의의 개별 노드가 임의의 다른 노드에 대한 시빌 사용자의 가능성을 판별하는 방식을 제안하였다. GateKeeper[7]는 네트워크 플로우 기반의 시빌 사용자 확인 기법이다. 이 기법은 임의의 개별 노드에서 일정한 용량을 가지는 네트워크 플로우를 시작 시키고 이 네트워크 플로우가 attack edge를 잘 지나지 못한다는 점과 지나더라도 제한된 용량의 플로우만이 지난다는 점을 이용한다. 하지만 이 연구들은 개별 판별자 중심의 연구 결과를 보여주고 있다. 개별 판별자를 사용할 경우, 노드 및 네트워크의 특성에 따라 시빌 판별 성능에 큰 편차가 생길 수 있다.

3. 시스템-차원 시빌-저항 신뢰도 추출 기법

이 논문에서 소개하는 시스템-차원 시빌-저항 신뢰도 추출 기법은 RRI(Random Route Intersection), RRTI(Random Route Tail Intersection), 그리고 RWTI(Random Walk Tail Intersection)이다. 이들 기법들은 전체 온라인 소셜 네트워크 그래프를 알 수 있는 서비스 제공자의 중앙 서버에서 수행된다고 가정한다.

각 기법들은 사용되는 세부 샘플링 기법에 따라서 구분되지만, 기본적인 샘플링 및 판별 방법은 모두 같다. 임의의 판별자 노드 v가 임의의 노드 s가 시빌 노드일 가능성의 정도를 판별하기 위해서는 우선 노드 v와 노드 s에서 소셜 네트워크 그래프 정보 샘플링을 수행하여 각 노드에 대한 샘플 모음을 준비한다. 준비된 두 샘플 모음들을 비교하여, 만약 최소한 하나 이상의 공통된 샘플이 두 샘플 모음에서 발견될 경우 판별자 노드 v가 임의의 노드 s를 "accept"했다고 판별한다. 이때 Birthday Paradox에서 제안하는 일정 수준 이상의 accept 확률을 얻기 위해서는 각 샘플 모음에 속한 샘플의 개수가 충분히 많아야 한다.

시스템-차원 시빌-저항 신뢰도를 추출하기 위해서 위와 같은 개별 노드에서 수행되는 판별기법을 다수의 판별자에서 수행한다. 이때 사용되는 다수의 판별자는 신뢰 할 수 있는 몇몇 노드에서 기존에 제안된 다른 기법들[5][6][7]중 하나를 사용하여 정직한 노드라고 판별된 노드들중에서 선택된다. 이 논문에서는 SybilLimit을 사용해 판별자 노드들을 선택하여 실험을 수행하였다. 이렇게 선택된 다수의 판별자에서 수행된 판별 기법의 결과를 종합한 시스템-차원 시빌-저항 신뢰도 T는 식 (1)과 같이 정의된다.

$$T = \frac{number\ of\ accepted\ verifiers}{number\ of\ all\ verifiers} \tag{1}$$

시스템-차원 시빌-저항 신뢰도 T는 0이상 1이하의 값을 가지고, 이 값이 1에 가까울수록 해당하는 노드가 정직한 사용자일 가능성이 높음을 의미한다.

3.1 Random Route Intersection

RRI(Random Route Intersection) 기법은 SybilGuard[5]에서 제안된 랜덤 라우트(Random Route)을 사용하여 샘플링을 수행한다. 랜덤 라우트는 변형된 랜덤 워크이다. 랜덤 워크는 각 노드에서 다음 노드를 랜덤하게 선택하는 반면에 랜덤 라우트는 각 노드에서 미리 정의된 들어오는 에지(incoming edge)와 나가는 에지(outgoing edge)의 일대일 맵핑 테이블(One-to-one mapping table)을 통해 다음 노드를 선택한다. 랜덤 라우트를 위한 맵핑 테이블의 엔트리 개수는 노드의 에지 개수와 같다. 임의의 에지를 통해 들어오는 랜덤 라우트들은 항상 지정된 에지를 통해 나가게 된다. 이에 따라 랜덤 라우트를 사용할 경우 convergence property를 기대할 수 있게 된다. 즉, 임의의 노드의 같은 에지를 통해서 들어오는 두 개의 랜덤 라우트는 항상 같은 에지를 통해서 나가게 된다.

RRI에서는 임의의 노드 v_i가 샘플링을 수행하기 위해 v_i의 모든 에지 $e_{ij} = \{e_{ij} | \forall v_j\ which\ is\ neighbor\ of\ v_i\}$에서 시작하는 길이가 w인 랜덤 라우트를 시작한다. 이때, 모든 랜덤 라우트들을 통해 방문했던 모든 노드들을 샘플 모음에

모은다. 길이가 w인 랜덤 라우트는 w번의 노드를 맵핑 테이블에 따라 방문하고 그 프로세스를 멈춘다. 따라서, 임의의 노드에서 RRI를 사용하여 수집하는 샘플 모음의 크기는 대략 각 노드의 평균 에지 개수에 w를 곱한 값이 된다.

사용하는 랜덤 라우트의 길이인 w는 RRI의 성능을 결정한다. 임의의 fast-mixing 특성을 가지는 그래프에서 사용할 수 있는 적절한 w의 값은 $\Theta(\sqrt{N}\log N)$이다.[5] w를 증가시키면, 샘플 모음에 모인 노드 샘플의 개수가 증가 하게 된다. 이에 따라, 임의의 판별자가 다른 노드를 accept할 확률이 높아지게 된다. 하지만 w값의 증가는 정직한 사용자 노드뿐만 아니라 시빌 사용자 노드가 accept될 확률을 증가시킨다. 또한 높은 w값은 랜덤 라우트가 attack edge를 지나갈 확률을 높여 시빌 사용자 노드가 accept될 확률을 크게 높일 수 있다.

3.2 Random Route Tail Intersection

RRI와 마찬가지로 RRTI에서도 랜덤 라우트를 사용하여 샘플링을 수행한다. 하지만, RRI에서 랜덤 라우트가 방문한 모든 노드를 샘플 모음에 추가한 것과 달리, RRTI는 랜덤 라우트의 마지막 direct 에지 즉 tail을 샘플 모음에 추가한다. 즉 RRI는 길이가 w인 랜덤 라우트 하나에서 w개의 샘플을 모으는 반면, RRTI는 랜덤 라우트 하나에서 1개의 샘플을 모으게 된다. RRTI에서는 임의의 fast-mixing 그래프에서 충분한 길이($\Theta(\log N)$)의 랜덤 라우트를 수행할 경우 그 tail은 그래프 상에서 균등하게 분포된다는 점을 이용한다.[6] 이때 주목할 점은 RRTI에서 사용하는 랜덤 라우트의 길이($\Theta(\log N)$)는 RRI에서 사용하는 랜덤 라우트의 길이($\Theta(\sqrt{N}\log N)$)보다 짧다는 것이다.

RRTI는 한 번의 랜덤 라우트를 수행할 때 1개의 샘플을 얻기 때문에, 서로 독립적인 균등 분포 특성을 가지는 r개의 샘플을 구하기 위해서는 각 노드에 r개의 랜덤 라우트용 맵핑 테이블이 필요하다. 사용하는 네트워크 그래프가 fast-mixing특성을 가지고 있다고 할 때, Birthday Paradox에 따라 임의의 노드를 판별하기 위해 필요한 샘플 모음의 크기인 r의 값은 대략 $\Theta(\sqrt{M})$이다.

RRTI 샘플링을 위해서는 r번의 랜덤 라우트를 수행한다. 이 때, $n(n<r)$번째 랜덤 라우트는 각 노드에 준비된 n번째 맵핑 테이블을 사용한다.

RRI에 비해 보다 짧은 길이의 랜덤 라우트를 사용해 샘플 모음을 준비하는 RRTI에서는 랜덤 라우트가 attack edge를 넘어갈 확률을 현저히 줄여준다. 하지만 RRTI는 정확한 동작을 위해 r개의 맵핑 테이블이 필요하게 된다. 즉 RRTI는 아주 큰 runtime 메모리를 필요로 한다. 또한 RRTI는 r번의 랜덤 라우트를 수행해주어야 하므로, RRI에 비해서 샘플링 수행 시간이 더 길어질 수 있다.

3.3 Random Walk Tail Intersection

RWTI는 RRI와 RRTI에서 사용하는 랜덤 라우트 대신 일반적인 랜덤 워크를 사용한다. 이는 랜덤 라우트를 수행하기 위해 준비해야 하는 맵핑 테이블 오버헤드를 줄이기 위함이다. 비록 랜덤 라우트를 사용하지 않음으로써 convergence property를 잃게 되지만, 충분한 길이($\Theta(\log N)$)의 랜덤 워크의 tail이 균등하게 분포되는 성질은 유지 된다.[8]

RRTI와 비슷하게, RWTI는 길이가 $w(=\Theta(\log N))$인 랜덤 워크를 수행 후 마지막 direct 에지인 tail을 샘플 모음으로 추가한다. 즉 한 번의 랜덤 워크는 하나의 샘플을 추가하기 때문에, RRTI와 비슷하게 $r(=\Theta(\sqrt{M}))$번의 랜덤 워크를 수행한다.

RRTI와는 달리 RWTI는 맵핑 테이블을 준비할 필요가 없기 때문에 샘플링 수행 시 필요한 메모리의 크기가 현저히 줄어든다. 그렇지만, RWTI에서 사용하는 랜덤 워크의 길이가 RRTI에서 사용하는 RRTI와 비슷하기 때문에 두 기법은 서로 비슷한 성능을 낸다.

4. 성능 평가

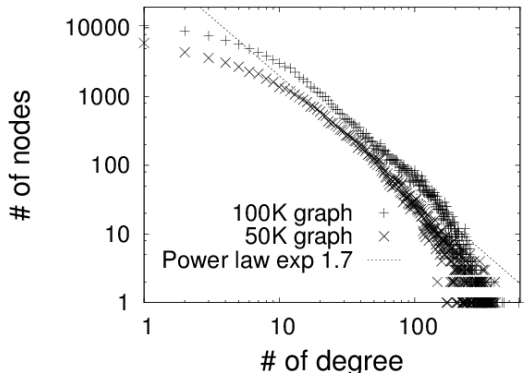

Fig. 1. Degree Distribution of Sampled Facebook Graphs

제안된 세 가지 타입의 시스템-차원 시빌-저항 신뢰도 추출 기법에 대한 성능을 평가하기 위해, Facebook에서 추출된 샘플 그래프[9]를 사용하였다. OSN 샘플 그래프는 Facebook에서 Forest-fire 샘플링 기법을 통해 추출하였고 각 그래프는 하나의 strongly connected component이다. 실험에 사용된 두 가지 샘플 그래프는 각각 노드의 개수가 50,000와 100,000이고 edge의 개수는 905,004와 1,861,360다. 이 두 그래프의 Diameter는 18로 같고 radius도 6로 같다. Fig. 1과 같이, 이 두 그래프는 계수를 약 1.7로 가지는 파워-로(Power-Law) 분산 특성을 보인다. 이 샘플 그래프는 honest region으로 가정하였다. 반면 Sybil region은 가상으로 그래프를 생성하여 구성하였다. 구성된 Sybil region은 다양한 개수의 시빌 사용자로 구성하였고 각 노드의 평균 에지 개수를 14로 설정하였다. 각 추출 기법에서 사용한 판별자 노드의 수는 100으로 설정하였다. 실험은 두 그래프를 사용하여 수행되었으나, 각 그래프에 대한 실험 결과가 아주 비슷하게 나온 관계로, 이 논문에서는 100,000개의 노드를 가지는 샘플 그래프에 해당하는 실험 결과를 중점적으로 기술한다.

4.1 초기 설정 실험 결과

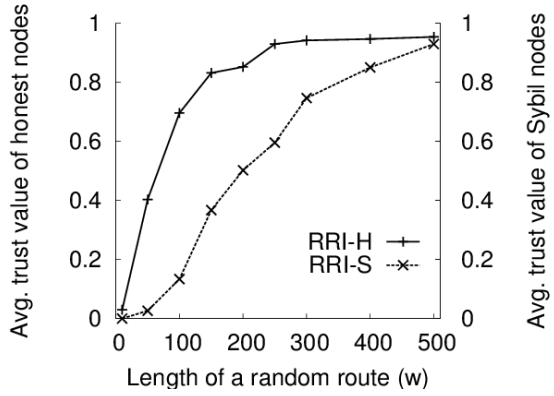

Fig. 2. Preliminary result - RRI

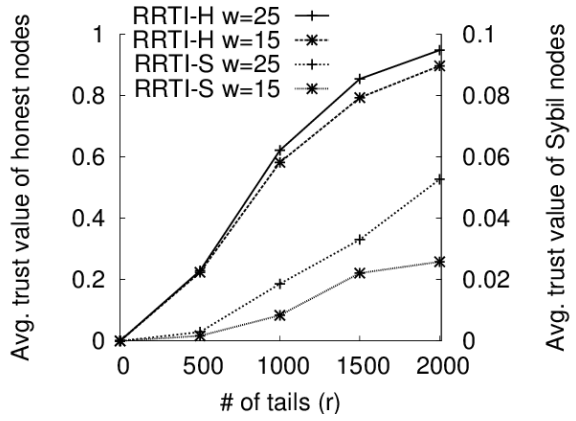

Fig. 3. Preliminary result - RRTI

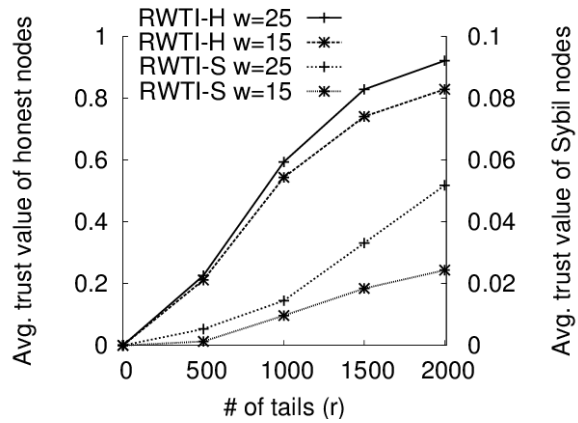

Fig. 4. Preliminary result - RWTI

우선, 각 추출 기법의 파라메터들(w,r)이 미치는 영향을 알아보고, 정직한 사용자들에게 일정 수준 이상의 높은 신뢰도를 책정할 수 있는 적정한 파라메터 값들을 찾기 위한 초기 설정 실험을 수행 하였다. 다양한 파라메터값들을 적용한 각 추출기법들에 대한 성능은 Fig. 2, Fig. 3, 그리고 Fig. 4에서 볼 수 있다. 이때 attack edge의 개수는 2로 설정하였다.

Fig. 2에서 RRI는 랜덤 라우트의 길이(w)에 크게 영향을 받는 것을 알 수 있다. 정직한 사용자들에게 평균 0.8이상의 신뢰도 값을 주기 위해서는 그 길이가 최소한 100이상은 되어야 한다. 하지만 그 길이가 길어짐에 따라 시빌 사용자들에게 주어지는 신뢰도 값도 거의 비례해서 커지는 것을 확인 할 수 있다.

Fig. 3과 Fig. 4에서 RRTI와 RWTI는 랜덤 라우트의 길이(w)보다는 샘플 모음의 크기(r)에 크게 영향을 받는 것을 알 수 있다. 샘플 모음의 크기가 커지면, 정직한 사용자의 평균 신뢰도 값이 비례하여 커진다. 시빌 사용자의 경우에도 샘플 모음의 크기가 커질수록 그 신뢰도가 증가하는 것을 볼 수 있지만, 주어지는 초기 신뢰도 및 증가분이 정직한 사용자가 받는 신뢰도 값에 비해 10배 이상 적은 것이 확인되었다.

Fig. 4에서는 RRTI와 비교해서 RWTI가 시빌 사용자에게 거의 비슷한 수준의 아주 낮은 신뢰도를 주는 반면, 정직한 사용자에게는 RRTI에서 주어지는 신뢰도 값보다 약간 낮은 신뢰도를 주는 것이 확인되었다. 랜덤 워크를 사용하면서 놓친 convergence property가 이 현상의 주된 원인으로 분석 된다. 만약 RWTI가 RRTI와 비슷한 수준의 신뢰도를 정직한 사용자들에게 제공하고자 한다면, 보다 긴 길이의 랜덤 워크를 사용해야 한다.

이 결과를 기준으로, 나머지 실험들은 RRI는 w를 200으로, RRTI는 w를 15, r을 2000으로, RWTI는 w를 20, r을 2000으로 설정하여 수행되었다.

4.2 Honest region의 신뢰도

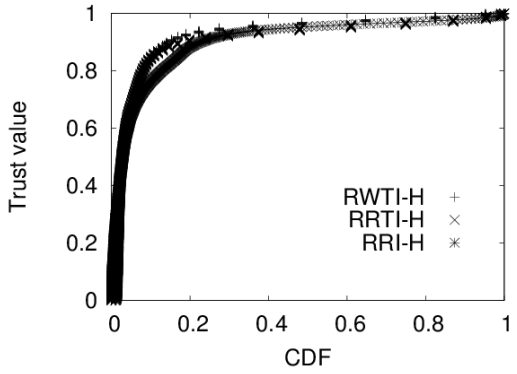

Fig. 5. Distribution of Honest users' trust value

우선 각 추출기법들이 honest region에 속한 노드들을 평가하는 성능을 측정하였다. Fig. 5에서는 각 추출기법들을 사용해 honest region에 속한 노드들에게 할당한 시스템-차원 시빌-저항 신뢰도 값의 분포를 Cumulative Distribution Function 형태로 나타내었다. 이 그림에서, RRTI와 RWTI를 사용할 경우, honest region의 약 90%정도는 0.8이상의 신뢰도를 가지는 것을 알 수 있는 반면, RRI를 사용할 경우 honest region의 약 85%정도만이 0.8 이상의 신뢰도를 가지는 것을 알 수 있다.

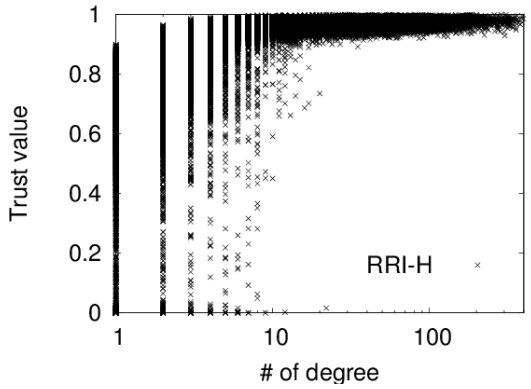

Fig. 6. Distribution of Honest users' trust value - RRI

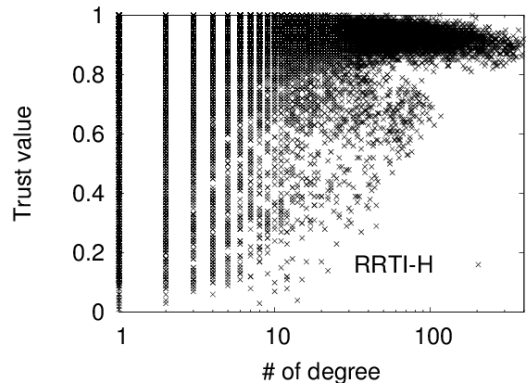

Fig. 7. Distribution of Honest users' trust value - RRTI

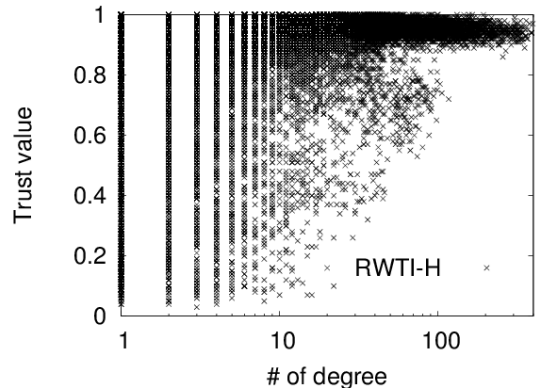

Fig. 8. Distribution of Honest users' trust value - RWTI

이러한 각 추출 기법들에 따른 honest region에 속한 노드들의 신뢰도 분포를 더욱 자세히 보기 위해 Fig. 6, Fig. 7, Fig. 8에서는 각 추출 기법들에 따른 신뢰도의 분포를 각 노드의 이웃노드 개수(degree)에 따라서 표현하였다. Fig. 6에서는 RRI를 사용할 경우 노드의 degree가 10보다 클 경우 신뢰도의 값이 0.9이상의 값이 되는 반면, 10 보다 작을 경우 아주 작은 신뢰도를 가질 수 있음을 확인할 수 있다. Fig. 7에서는 RRTI를 사용할 경우 RRI의 결과와 비슷하게 노드의 degree가 10보다 커야 큰 값의 신뢰도를 가질 수 있음을 알 수 있다. 또한 RRTI의 경우에는 노드의 degree가 100보다 작을 경우에는 중간값(0.3~0.8) 정도의 trust value를 가질 수 있음을 확인하였다. Fig. 8은 Fig. 7과 비슷한 분포를 보이는데, 이에 따라 RWTI와 RRTI가 비슷한 특성을 가진다고 말할 수 있다.

4.3 Sybil region의 신뢰도

각 추출 기법들이 Sybil region에 속한 시빌 사용자들에 대한 신뢰도를 추출하는 성능을 평가하기 위해, Sybil region과 honest region사이의 attack edge의 개수를 변화시켜가며 시빌 사용자들의 신뢰도를 측정하였다. Fig. 9에서 각 추출기법들에 의해 얻어진 시빌 사용자들의 신뢰도 값의 평균값을 attack edge의 개수에 따라서 나타내었다. 이 그림에서 RRI기법의 경우, attack edge가 20개 정도만 되더라도 시빌 사용자들의 신뢰도 평균값이 0.8이상이 됨을 확인하였다. 즉, RRI기법으로 추출한 신뢰도는 attack edge에 심각하게 영향을 받는 것으로 확인되었다. 반면, RRTI와 RWTI의 경우, attack edge가 200개 정도 될 경우 평균 0.6의 신뢰도를 시빌 사용자들에게 제공하는 것을 확인하였다. 즉, 상대적으로 RRI기법보다는 RRTI와 RWTI기법이 attack edge의 개수에 덜 민감하게 반응함을 확인하였다. 또한 RRTI가 RWTI보다는 미세하게나마 attack edge의 영향을 덜 받는 것으로 확인되었고, convergence property의 부재가 그 원인으로 분석 된다.

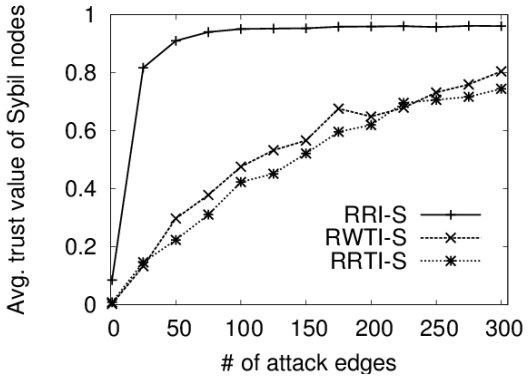

Fig. 9. Distribution of Sybil users' trust value as a function of the number of attack edges

이러한 attack edge의 각 추출 기법들에 대한 영향을 보다 자세히 알아보기 위해 Fig. 10, Fig. 11, Fig. 12에서는 각 추출 기법들을 사용해서 얻어진 시빌 사용자들의 신뢰도 분포를 Cumulative Distribution Function의 형태로 표현하였다. Fig. 10에서는 RRI를 사용할 경우 attack edge의 개수가 10일 경우 시빌 사용자의 40%가 0.6 이상의 신뢰도를 가지게 되고, attack edge의 개수가 50일 경우 시빌 사용자의 90%가 0.8 이상의 신뢰도를 가지게 되는 것을 확인하였다. 즉 RRI의 경우 attack edge를 포함하는 시빌 사용자 근처 노드들은 높은 신뢰도를 가지게 됨으로써 attack edge의 개수가 늘어남에 따라 급속도로 신뢰도의 평균값이 증가하게 된다. RRTI를 사용할 경우에도 Fig. 11과 같이 attack edge

가 증가할 때 전체 시빌 사용자들의 신뢰도 평균값은 증가한다. 하지만, RRI의 경우와 달리 RRTI의 경우에는 attack edge의 개수가 증가함에 따라 모든 시빌 사용자들의 신뢰도가 서로 비슷한 값을 가지면서 증가하는 것을 확인하였다. 즉 attack edge가 300일 경우, RRI의 경우 대부분의 시빌 노드는 0.9 이상의 아주 큰 값의 신뢰도를 가지는 반면, RRTI의 경우 대부분의 시빌 노드들은 약 0.7정도의 신뢰도를 가지고 극소수의 노드들이 최대 0.8 정도의 신뢰도를 가지는 것을 확인하였다. 즉, honest region에 속한 노드의 신뢰도 평균값이 0.8 이상인 것을 고려하면, RRTI의 경우 attack edge의 개수가 300정도로 늘어나더라도, 여전히 시빌

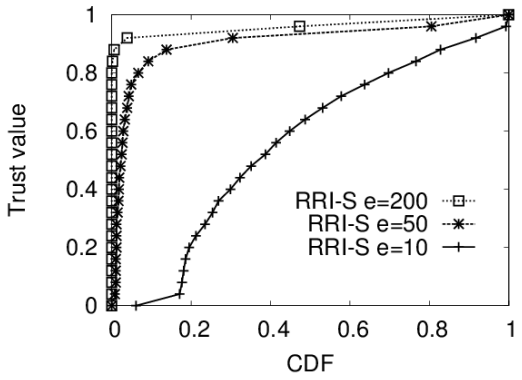

Fig. 10. Distribution of Sybil users' trust value - RRI

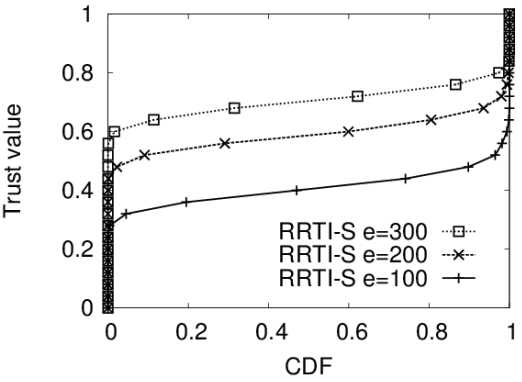

Fig. 11. Distribution of Sybil users' trust value - RRTI

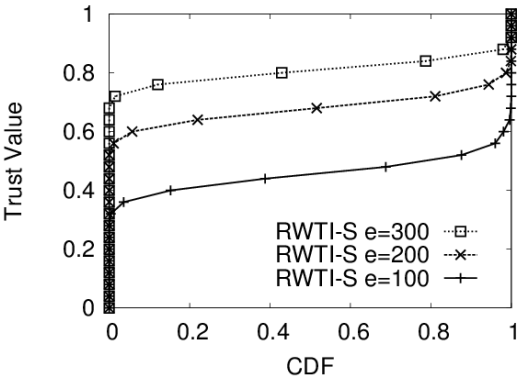

Fig. 12. Distribution of Sybil users' trust value - RWTI

사용자들은 상대적으로 낮은 신뢰도를 할당 받게 된다. RWTI의 경우도 Fig. 12에서 나타낸 것과 같이 RRTI와 비슷한 특성을 가지는 것을 알 수 있다.

4.4 시빌-저항 신뢰도의 Kazza

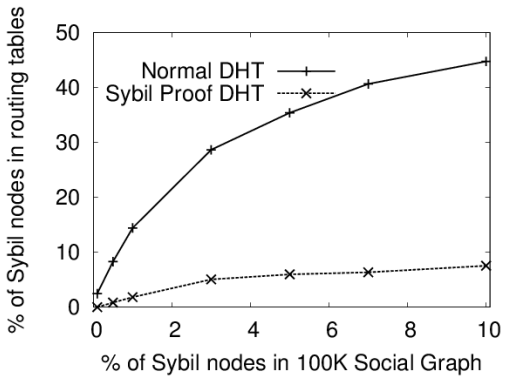

Fig. 13. Impact of system-wide Sybil-resistant trust value to Kademlia DHT

제안된 추출 기법을 통해 얻어지는 시스템-차원 시빌-저항 신뢰도는 Reputation시스템[10], 스팸 필터링 시스템[9], DHT기반 P2P시스템등 다양한 분산 시스템에 사용될 수 있다. Fig. 13은 Kademlia DHT[11]를 사용하는 P2P 파일 공유 시스템에서 시스템-차원 시빌-저항 신뢰도가 미치는 영향에 대한 실험 결과이다. 이 실험에서 사용된 시빌 노드들은 DHT시스템을 전복시키기 위한 이클립스 어택(Eclipse Attack)을 수행하도록 동작한다. 이클립스 어택이란 DHT 기반의 P2P시스템에서 각 노드의 DHT를 시빌 노드들로 채워 해당 P2P시스템의 메시지 전달 제어권을 뺏거나 메시지 전송을 불가능하게 하는 행위이다. 일반적으로 DHT테이블의 30%이상이 시빌 사용자로 채워지게 되면 이 P2P시스템은 이클립스 어택에 의해 시스템이 망가졌다고 볼 수 있다.

Fig. 13에서 시스템-차원 시빌-저항 신뢰도를 사용하지 않았을 경우 P2P시스템에 참여하는 노드중 3%가 시빌 노드일 경우 전체 DHT엔트리의 30%가 시빌 노드 정보로 채워짐을 확인 하였다. 반면, 신뢰도를 사용한 Sybil-Proof DHT의 경우에는 시빌 노드가 전체 노드의 10%이상이더라도 DHT엔트리의 9% 미만이 사용되는 것을 확인 할 수 있었다. 이 실험 결과를 통해 분산 시스템에서 시스템-차원 시빌-저항 신뢰도를 사용하지 않을 경우 시빌 어택에 의해 시스템이 쉽게 망가질 수 있다는 점과 시스템-차원 시빌-저항 신뢰도가 분산 시스템을 안정적으로 운용하는데 있어 아주 중요하다는 점을 확인하였다.

5. 결 론

익명의 사용자들에 의해서 운용되는 분산시스템을 시빌 어택에 강인하게 설계하는 것은 중요하다. 이 논문에서는 온라인 소셜 네트워크 그래프에 내포된 시스템-차원 시빌-

저항 신뢰도를 추출하는 기법들을 제안하고 그 성능을 실험을 통해 확인하고 분석하였다. 성능평가를 통해 각 추출 기법들에서 사용되는 파라메터들(랜덤 라우트/워크 길이, 샘플 모음의 크기)이 미치는 영향을 분석할 수 있었다. 이중 RRI가 가장 빠른 수행속도를 보였지만, attack edge의 변화에 너무 심각하게 영향을 받기 때문에 적절한 신뢰도를 할당하기에는 무리가 있음을 확인하였다. RRTI와 RWTI 사이에는 성능과 처리 속도간의 비교점을 찾을 수 있었다. RRTI와 RWTI의 신뢰도 추출 성능은 아주 유사하나 attack edge의 영향 측면에서 RRTI가 좀 더 우수함을 확인 하였다. 하지만, 메모리 사용량 및 처리속도 측면에서는 RWTI가 RRTI보다 우수함을 확인하였다.

또한 제안된 추출 기법들을 통해 높은 (0.8~) 신뢰도를 얻기 위해서는 온라인 소셜 네트워크 그래프 상에서 이웃노드가 최소한 10개 이상 되어야 하고, RRTI나 RWTI의 경우에는 이웃노드가 100개 이상 되어야 0.8 이상의 신뢰값을 가질 수 있음을 확인 하였다.

이러한 추출 기법들을 통해 얻어지는 시스템-차원 시빌-저항 신뢰도는 다양한 분산 시스템에 적용가능하며, 시빌 어택을 효과적으로 막을 수 있다.

참 고 문 헌

[1] S. D. Kamvar, M. T. Schlosser and H. Garcia-molina. The EigenTrust Algorithm for Reputation Management in P2P Networks. In Proc. of 12th WWW, 2003.

[2] A.G.P. Rahbar and O. Yang. PowerTrust: A Robust and Scalable Reputation System for Trusted Peer-to-Peer Computing. In IEEE TOPDS, Vol.18, Issue.4, April, 2007.

[3] J. R. Douceur. The sybil attack. In Proceedings of IPTPS 2002, pp.251-260, 2002.

[4] Luis von Ahn, Manuel Blum, Nicholas J. Hopper and John Langford. CAPTCHA: Using Hard AI Problems for Security. In Proc. EUROCRYPT 2003, Springer LNCS Vol.2656, Warsaw, Poland.

[5] H. Yu, M. Kaminsky, P. B. Gibbons and A. Flaxman. SybilGuard: Defending Against Sybil Attacks via Social Networks. In Proc. of ACM SIGCOMM, August, 2006.

[6] H. Yu, P. B. Gibbons, M. Kaminsky and F. Xiao. SybilLimit:A Near-Optimal Social Network Defense against Sybil Attacks. In Proc. of IEEE Symposium on Security and Provacy 2008, pp.3-17, 2008.

[7] Nguyen Tran, Jinyang Li, Lakshiminarayanan Subramanian, and Sherman S.M. Chow. Optimal Sybil-resilient node admission control. In Proc. INFOCOMM 2011

[8] M. Mitzenmacher and E. Upfal. Probability and Computing. Cambridge University Press, 2005.

[9] Michael Sirivianos, Kyungbaek Kim and Xiaowei Yang. SocialFilter: Introducing Social Trust to Collaborative Spam Mitigation. In Proc. of IEEE INFOCOM 2011, April 10-15, 2011, Shanghai, China.

[10] Michael Sirivianos, Kyungbaek Kim, Jian Wei Gan and Xiaowei Yang. Assessing the Veracity of Identity Assertions via OSNs. In Proc. of COMSNETS 2012, January 3-7, 2012, Bangalore, India.

[11] Petar Maymounkov and David Mazieres, Kademlia: A Peer-to-peer Information System Based on the XOR Metric. In Proc. of 1st International Workshop on Peer-to-peer Systems (IPTPS'02), 2002.

[12] Kyungbaek Kim. Using multiple verifiers to detect Sybils in a social network graph. In Proc. of the 8th International Conference on Future Information Technology (FutureTech 2013), September 4-6, 2013, Gwangju, South Korea.

[13] Kyungbaek Kim. Assessing the impact of properties of a node to OSN based Sybil detection. In Proc. of the International Conference on Computer Applications and Information Processing Technology (CAIPT 2013), June 27-29, 2013, Prague, Czech Republic.

[14] Kyungbaek Kim. Sybil-Resistant Trust Value of Social Network Graph. In Proc. of the First International Conference on Smart Media and Applications (SMA 2012), August 21-24, 2012, Kunming, Yunnan, China.

김 경 백

e-mail : kyungbaekkim@chonnam.ac.kr

1999년 한국과학기술원 전자전산(학사)

2001년 한국과학기술원 전자전산(석사)

2007년 한국과학기술원 전자전산(박사)

2007년~2011년 University of California, Irvine, 박사후연구원

2012년~현 재 전남대학교 전자컴퓨터공학부 조교수

관심분야 : 분산시스템, 미들웨어, 피어투피어 네트워크, 오버레이 네트워크, 소셜 네트워크, 모바일 클라우드 시스템