NC STATE UNIVERSITY

# CSC 495.002 – Lecture 4
# Web/Social Networks Privacy: Violations and Regret

Dr. Özgür Kafalı

North Carolina State University
Department of Computer Science

Fall 2017

NC STATE UNIVERSITY



## Sharing and Disclosure

- Common usage scenarios of OSNs
- Common sharing and disclosure patterns of users
  - What content types are shared?
  - Whom are they shared with?
  - How do sharing behaviors change over time?
- Does shared content match intended audience?
- How do users mitigate privacy concerns?

## Problem Definition

- Violation: Reality does not meet user expectation about privacy
  - Mismatch between intended and actual audience
  - Unawareness of social links
- Regret: Later become unhappy about negative consequences of sharing behavior
  - Enumerate reasons to share
  - Identify regrettable actions
  - Help users avoid such actions

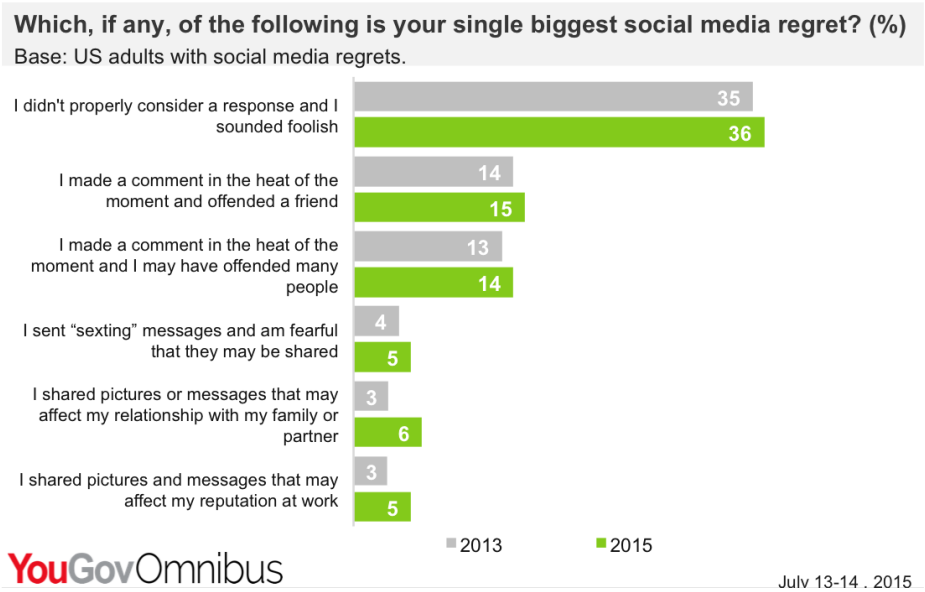




## Exercise: Regrettable Actions

- I shared pictures or messages that may affect my relationship with my family or partner
- I shared pictures or messages that may affect my reputation at work
- I made a comment in the heat of the moment and I may have offended many people
- I sent “sexting” messages and am fearful that they may be shared
- I made a comment in the heat of the moment and offended a friend
- I didn’t properly consider a response and I sounded foolish

# Regrettable Actions




http://www.huffingtonpost.com/shane-paul-neil/more-than-half-of-america_b_7872514.html

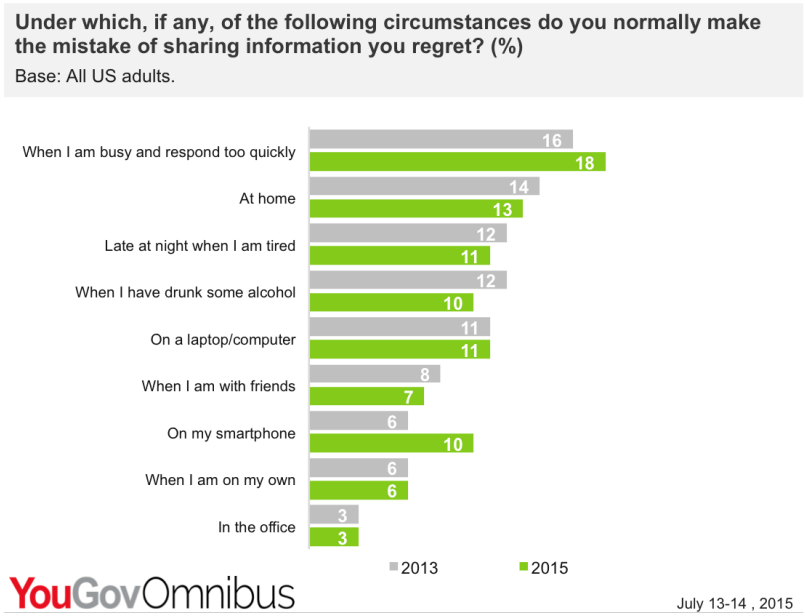




# Exercise: Reasons for Regret

- When I am with friends
- On a laptop/computer
- On my smartphone
- When I have drunk some alcohol
- At home
- Late night when I am tired
- When I am on my own
- When I am busy and respond too quickly
- In the office

# Reasons for Regret




http://www.huffingtonpost.com/shane-paul-neil/more-than-half-of-america_b_7872514.html





# Violation Types

- Norm violations
  - Norms describe normal (expected) behavior
  - Some norm violations are desirable (to maintain functionality)
- Violations of privacy laws
  - Some norms can be implemented as laws
  - Sanctions applied in case of violations
- Exceptions
  - Depends on user expectations
  - Not all violations are exceptions
  - There might be exceptions even if no violations

# Studies

- Look at two studies
  - One formal reasoning method for predicting privacy violations
  - One empirical study about regrets





# Detecting and Predicting Privacy Violations in Online Social Networks

**Detecting and Predicting Privacy Violations in Online Social Networks**

**Özgür Kafalı · Akın Günay · Pınar Yolum**

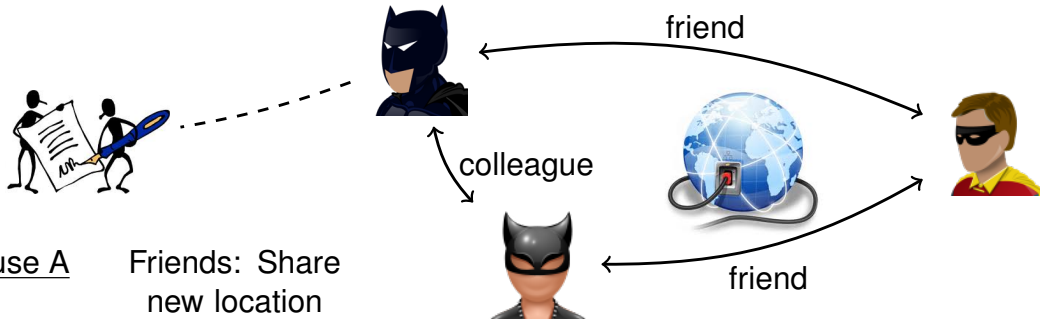
Received: date / Accepted: date

**Abstract** Online social networks have become an essential part of social and work life. They enable users to share, discuss, and create content together with various others. Obviously, not all content is meant to be seen by all. It is extremely important to ensure that content is only shown to those that are approved by the content's owner so that the owner's privacy is preserved. Generally, online social networks are promising to preserve privacy through privacy agreements, but still everyday new privacy leakages are taking place. Ideally, online social networks should be able to manage and maintain their agreements through well-founded methods. However, the dynamic nature of the online social networks is making it difficult to keep private information contained.

We have developed $\mathcal{PROTOSS}$, a run time tool for detecting and predicting $\mathcal{PR}$ivacy vi$\mathcal{O}$la$\mathcal{T}$ions in $\mathcal{O}$nline $\mathcal{S}$ocial network$\mathcal{S}$. $\mathcal{PROTOSS}$ captures relations among users, their privacy agreements with an online social network operator, as well as domain-based semantic information and rules. It uses model checking to detect if relations among the users will result in the violation of privacy agreements. It can further use the semantic information to infer possible violations that have not been specified by the user explicitly. In addition to detection, $\mathcal{PROTOSS}$ can predict possible future violations by feeding in a hypothetical future world state. Through a running example, we show that $\mathcal{PROTOSS}$ can detect and predict subtle leakages, similar to the ones reported in real life examples. We study the performance of our system on the scenario as well as on an existing Facebook dataset.

Kafalı et al. Detecting and Predicting Privacy Violations in Online Social Networks. Distributed and Parallel Databases, 32(1):161–190, 2014

# Privacy Contracts




Clause A Friends: Share new location

Clause B Colleagues: Do not disclose location

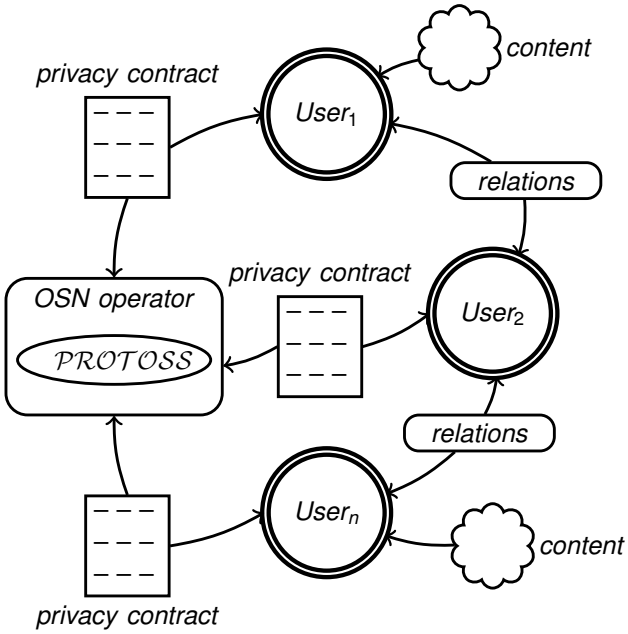
- What happens when Batman checks in at Arkham Asylum, who should know about that?
- What happens when Robin posts a picture together with Batman?



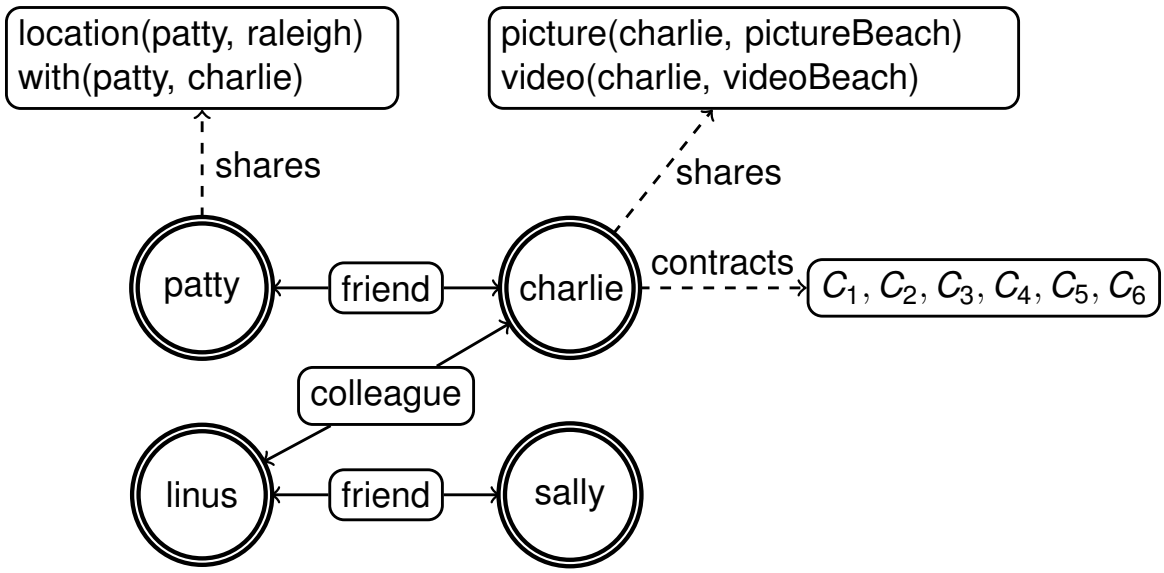


# *PROTOSS*: Contract-based OSN Architecture

- **PR**ivacy vi**O**la**T**ions in **O**nline **S**ocial network**S**



# Sharing Example

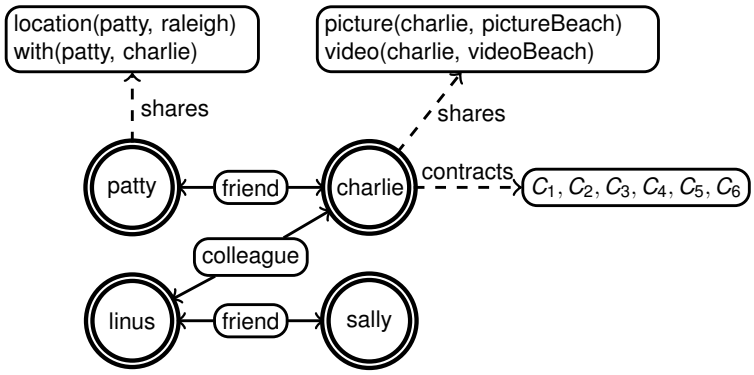






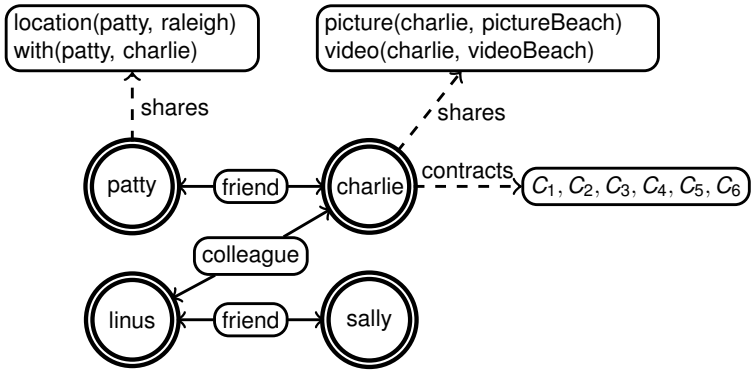

# Users

- Charlie
- Patty
- Sally
- Linus



# Relations

- *friend*(*X*, *Y*): Users *X* and *Y* are friends
- *colleague*(*X*, *Y*): Users *X* and *Y* are colleagues
- *friend*(*patty*, *charlie*)
- *friend*(*linus*, *sally*)
- *colleague*(*charlie*, *linus*)

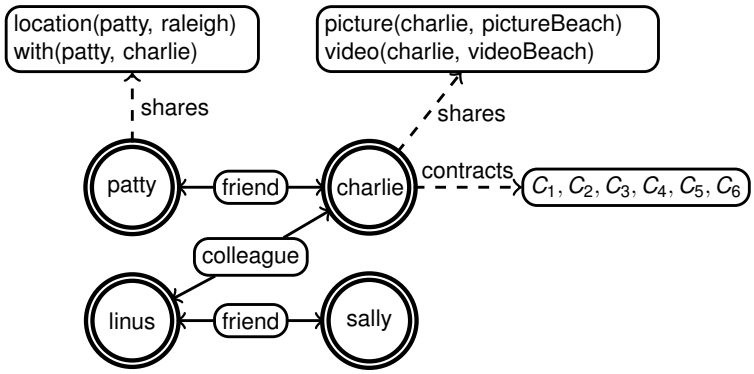







# Content

- *location*(*X*, *L*): User *X* is at location *L*.
- *with*(*X*, *Y*): User *X* is with user *Y*.
- *picture*(*X*, *P*): User *X* posts a picture *P*.
- *video*(*X*, *V*): User *X* posts a video *V*.



## OSN Behavior

- $B_1$: $visible(with(X, Y), Z) \leftarrow friend(X, Z) \vee friend(Y, Z)$
- $B_2$: $visible(location(X, L), Y) \leftarrow friend(X, Y)$
- $B_3$: $visible(picture(X, I), Y) \leftarrow friend(X, Y)$
- $B_4$: $visible(video(X, V), Y) \leftarrow friend(X, Y)$

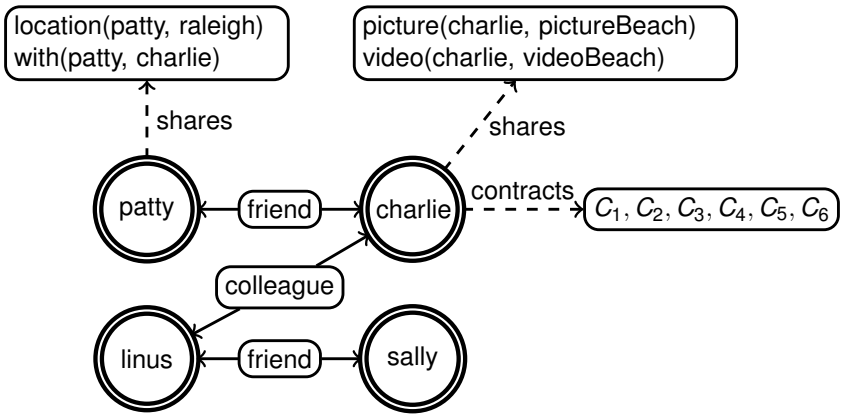




## OSN Contracts

- $C_1(osn, charlie, friend(charlie, Y), show(pic(charlie, P), Y))$
- $C_2(osn, charlie, friend(charlie, Y), show(with(charlie, Z), Y))$
- $C_3(osn, charlie, friend(charlie, Y), show(loc(charlie, L), Y))$
- $C_4(osn, charlie, colleague(charlie, Y), \neg show(pic(charlie, P), Y))$
- $C_5(osn, charlie, colleague(charlie, Y), \neg show(with(charlie, Z), Y))$
- $C_6(osn, charlie, colleague(charlie, Y), \neg show(loc(charlie, L), Y))$

# Scenario 1

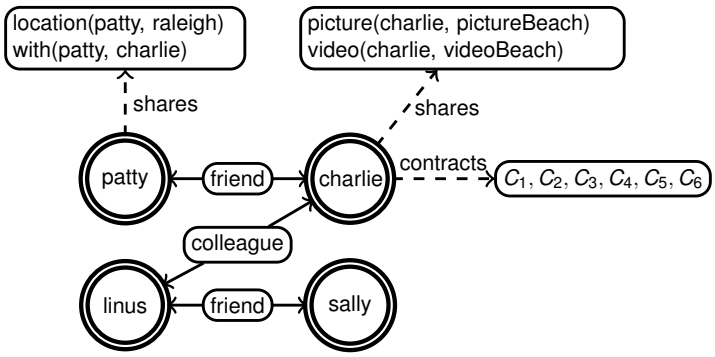



- According to contract $C_4$, pictures of *charlie* should not be revealed to his colleagues
- *linus* should not be able to see *charlie*'s picture *pictureBeach*





# Scenario 2




- According to contract $C_6$, location of *charlie* should not be revealed to his colleagues
- *linus* should not be able to see *charlie*'s location
- *charlie* does not share his location, but *patty* does (indirectly),
- She shares that she is with *charlie* and she is in Raleigh
- Inference: *charlie* is in Raleigh too

# Scenario 3

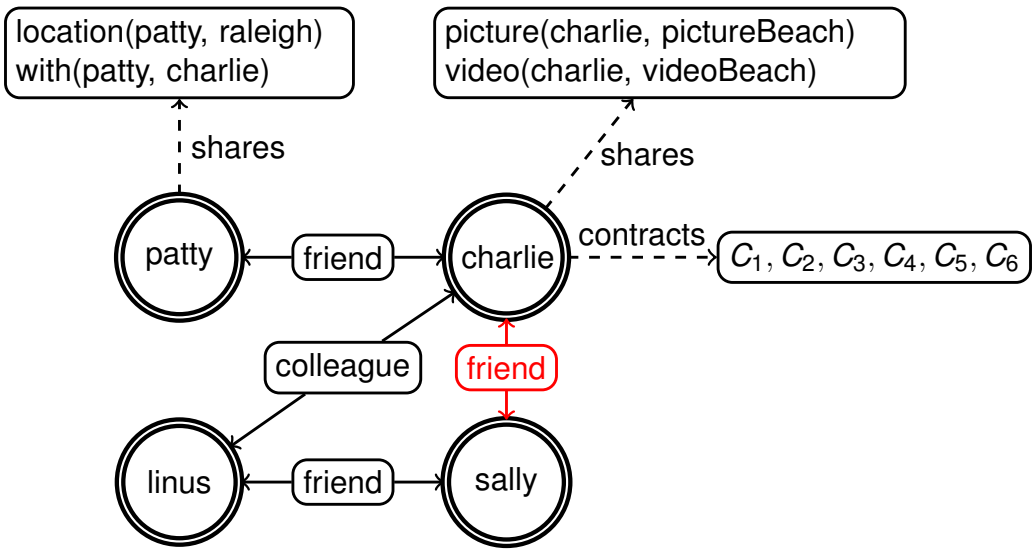
- As *charlie* stated in his privacy agreement, he does not want his colleagues to view his pictures (contract $C_4$)
- However, he has not made any statement about his videos (knowingly or unknowingly)
- Is it possible to make further reasoning to infer that videos are by nature similar to pictures?
- If any videos of *charlie* are being seen by colleagues, is it worthwhile to notify him?





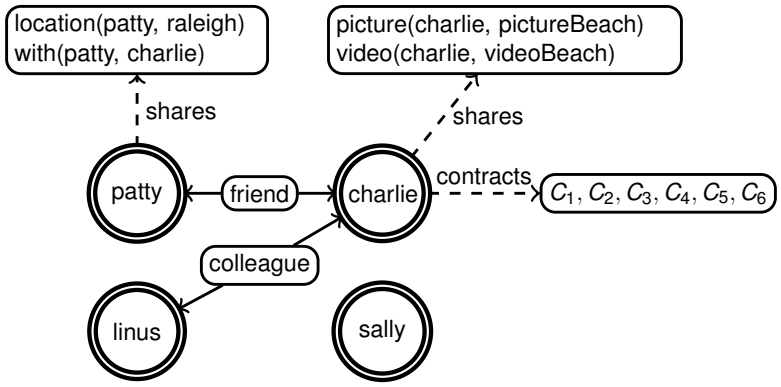
# Scenario 4

- Assume that *charlie* meets *sally* in Raleigh and adds her as a friend. Hence, OSN evolves into a new state. The aim is to detect whether *charlie*'s picture is visible to *linus*?



# Prediction Scenarios

- Go back to the initial state of the OSN, i.e., *charlie* and *sally* are not friends yet
- Look at the OSN from *charlie*'s point of view
- *charlie* tries to predict possible future breaches of his privacy depending on the evolution of relations between the users



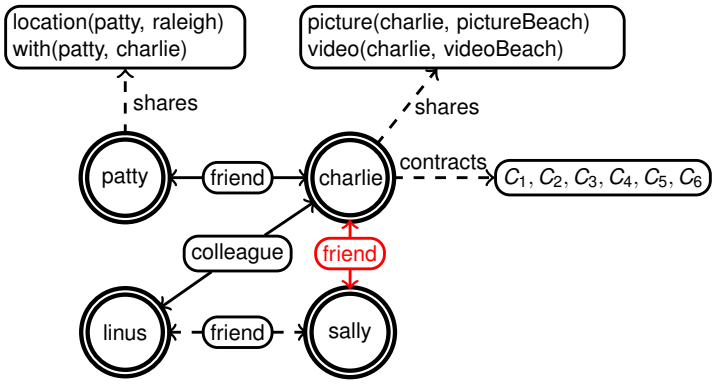





# Scenario 5

- *charlie* is a cautious user and desires to find out what would it take for *linus* to see his pictures
- That is, what relations in the OSN need to be initiated between the users of the OSN in the future for this information to leak?
- *charlie* chooses not to make any assumptions about the relations of the other users

# Scenario 6

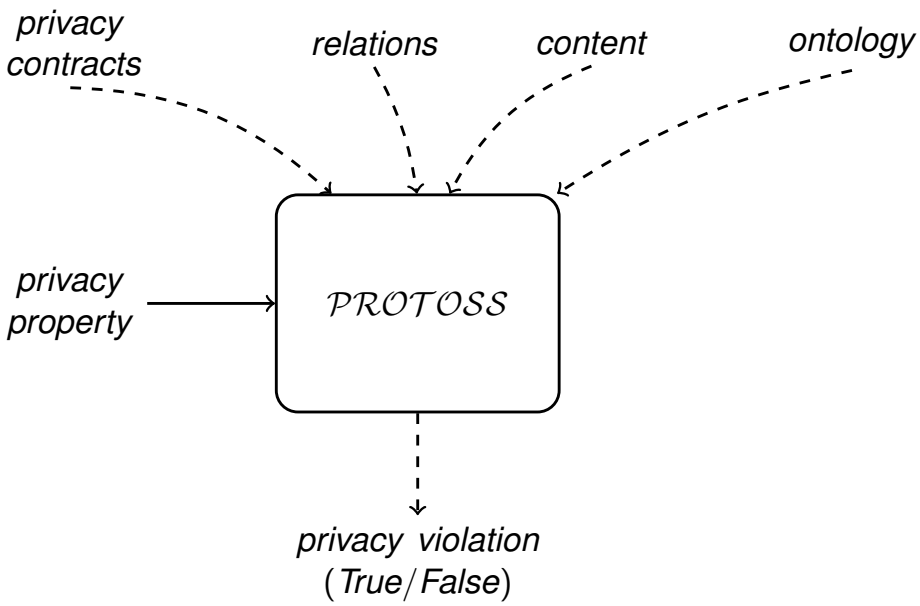



- *charlie* wants to add *sally* as a friend
- He is concerned that this may cause *linus* to see his pictures
- Before adding *sally* as a friend, he wants to find out whether his pictures would be visible to *linus* if he adds *sally* as a friend
- *charlie* assumes *sally* and *linus* are friends
- *charlie* assumes *patty* and *linus* are not friends

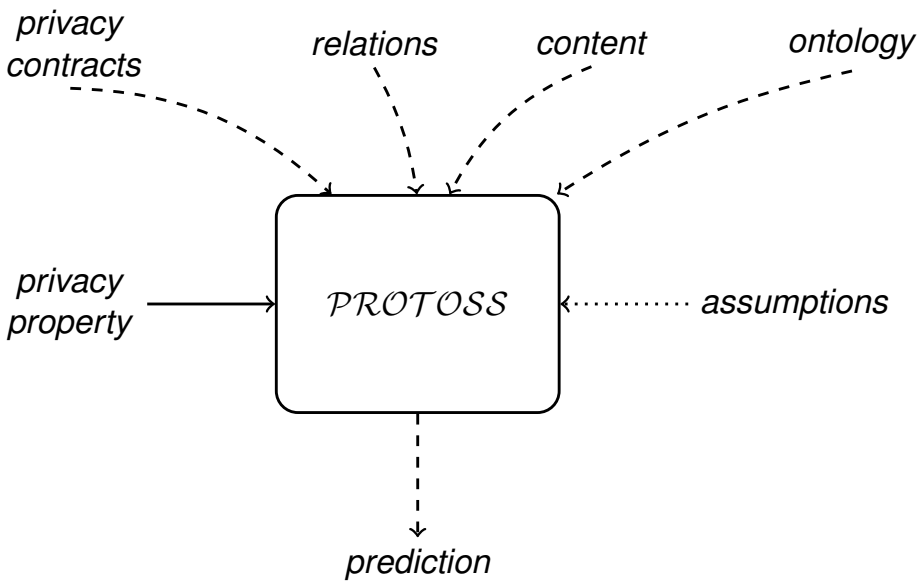




# Detecting Violations



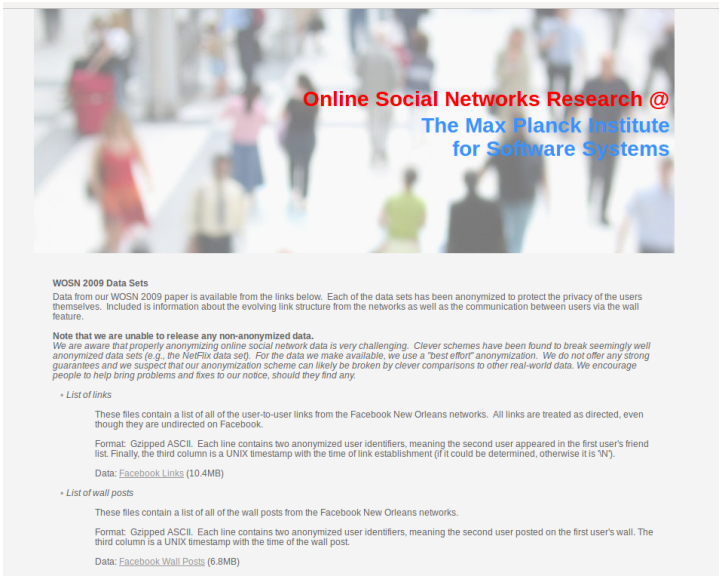
# Predicting Violations








# Facebook Dataset




- Alan Mislove's OSN dataset: http://socialnetworks.mpi-sws.mpg.de/data-wosn2009.html

# Dataset Details

| User | User | Timestamp |
|---|---|---|
| 1 | 18 | N |
| 1 | 20 | 1217964960 |
| 1 | 23 | N |
| 1 | 24 | 1227241074 |

- Each row lists two individuals that are related to each other
- Optionally a date that implies when the relationship between the two individuals were formed
- Does not contain different type of relations or contents
- Assumptions
  - Relations between individuals are friend relations
  - OSN will show the content posted by users (e.g., pictures) to their friends (not anyone else)
  - Users can repost contents initially posted by friends





# Methodology

- Research question: Is it possible for a user $Y$ to actually view a content posted by $X$, even though $X$ and $Y$ are not friends?
- Research question: If so, can we predict it before it happens?
- Take a subset of the dataset such that we begin with one user and add all of her friends and her friends' friends
- Previous work on link prediction has shown that it is very likely for a new friend to be already contained in the friends of friends network

## Violation Scenario

- Violation condition: $Y$ ends up viewing a content of $X$
- There exists a $Z$ that is both friends with $X$ and $Y$
- $Z$ shares the content of $X$ with $Y$
- OSN's commitment to $X$ is violated





## Performance Results

| User | #Users | #Friends | #States | Prediction time |
|---|---|---|---|---|
| 1 | 27 | 2129 | 894.4 K | 1.75 s |
| 163 | 26 | 1222 | 396.2 K | 0.94 s |
| 1645 | 29 | 679 | 127.1 K | 0.89 s |
| 31720 | 50 | 2294 | 557.6 K | 1.87 s |
| 48696 | 16 | 495 | 144.1 K | 0.50 s |

## Limitations

- Scalability: Model checking is a computationally expensive approach
- Relaxation of some of the assumptions: Beyond friend of friend





## In a Mood? Call Center Agents Can Tell

- News article: http://www.nytimes.com/2013/10/13/business/in-a-mood-call-center-agents-can-tell.html
- Links are also on the course website

# Things to Look For

- Root cause: What went wrong?
- If it was not intentional, what was the original aim?
- Affected parties
- Implications and similar problems
- Mitigation (using methods we have seen): Prevention, detection, recovery

- Take 10 minutes to look at the incident on your own

- Now discuss with your neighbor
- Also take a look at the summary report: https://drive.google.com/file/d/0B3m-I0YVAv0EcXlINGN6akI2M2M/view