Expressive Privacy Control with Pseudonyms

Seungyop Han, Vincent Liu, Qifan Pu, Simon Peter,
Thomas Anderson, Arvind Krishnamurthy, David
Wetherall

University of Washington

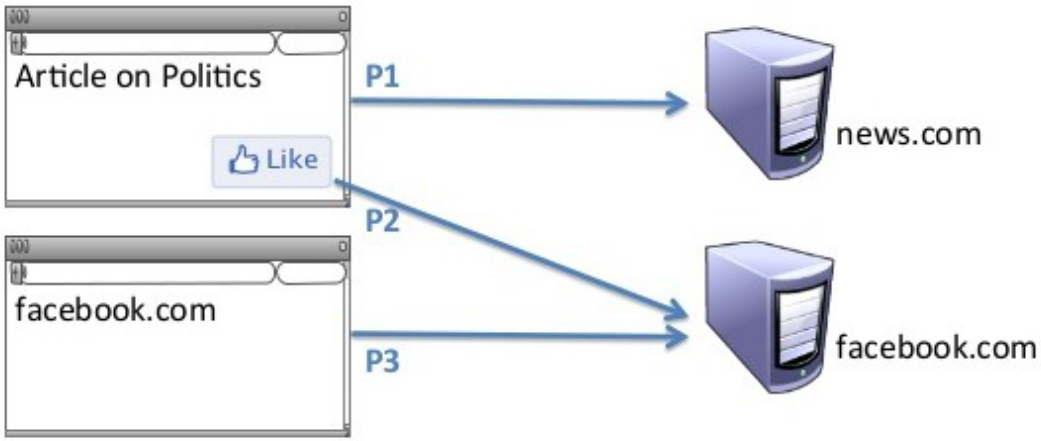
Expressive Privacy Control with Pseudonyms

Seungyop Han, Vincent Liu, Qifan Pu, Simon Peter,
Thomas Anderson, Arvind Krishnamurthy, David
Wetherall

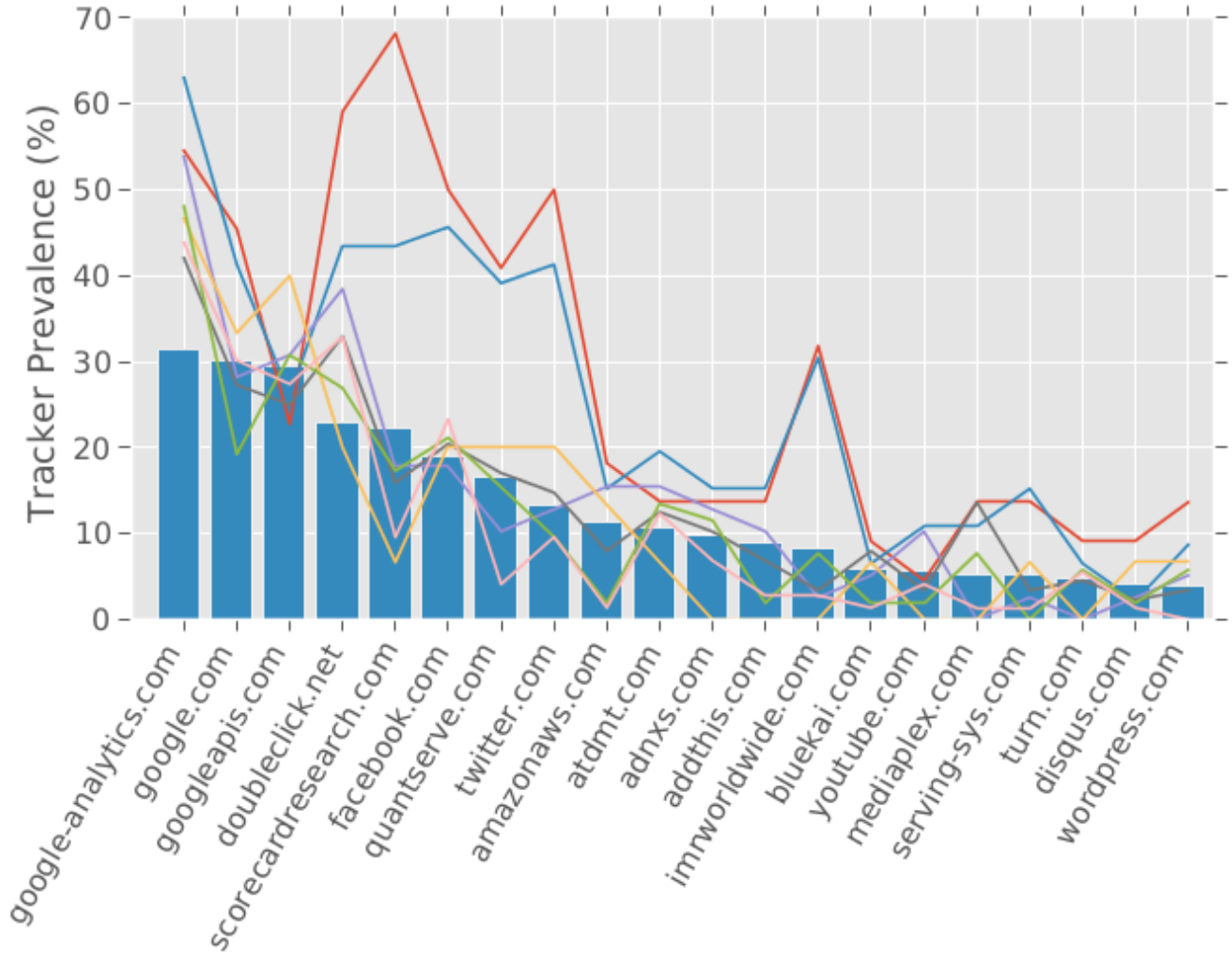
University of Washington

slides based on Seungyop Han's SIGCOMM presentation

Tracking

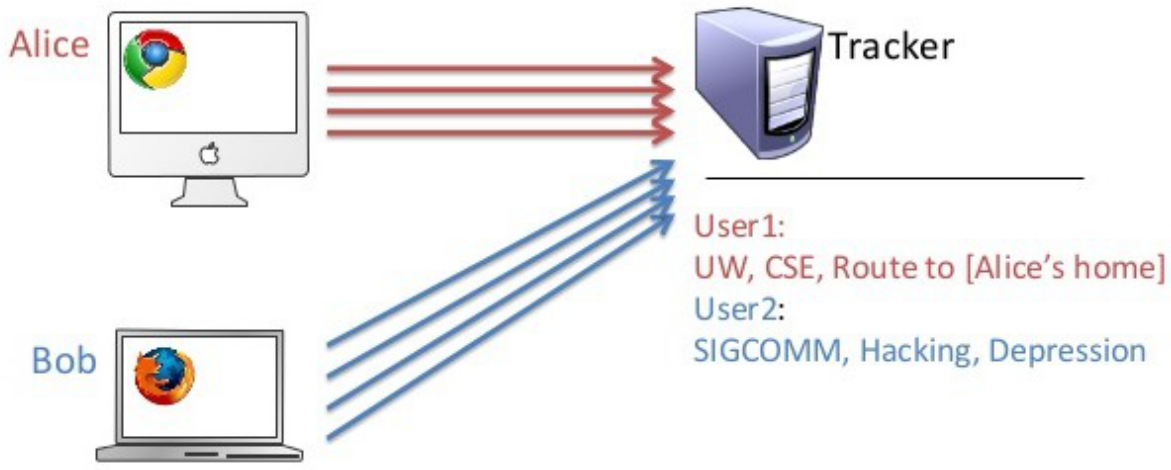

Article on Politics
Like
P1
news.com
P2
facebook.com
P3
facebook.com

Tracking is Pervasive



Tracking is Bad for Privacy

- Trackers can correlate activities the user might not want associated: i.e., business and sensitive personal information



Tracking is also Good?

- Banking websites track users to prevent fraud
- Targetted advertisements are better for users
- Economic engine of much of the internet

Tracking is opaque

- The user has no control over how they are tracked

What do Trackers track?

- **IP Address**
- **Application information**
  - Cookies
  - HTML5 LocalStorage
- **DNS**
- **System Information**

Current Solutions

- Application Layer
  - Block Third-party Cookies
  - DoNotTrack header

Can't handle lower-level information: IP address

Current Solutions

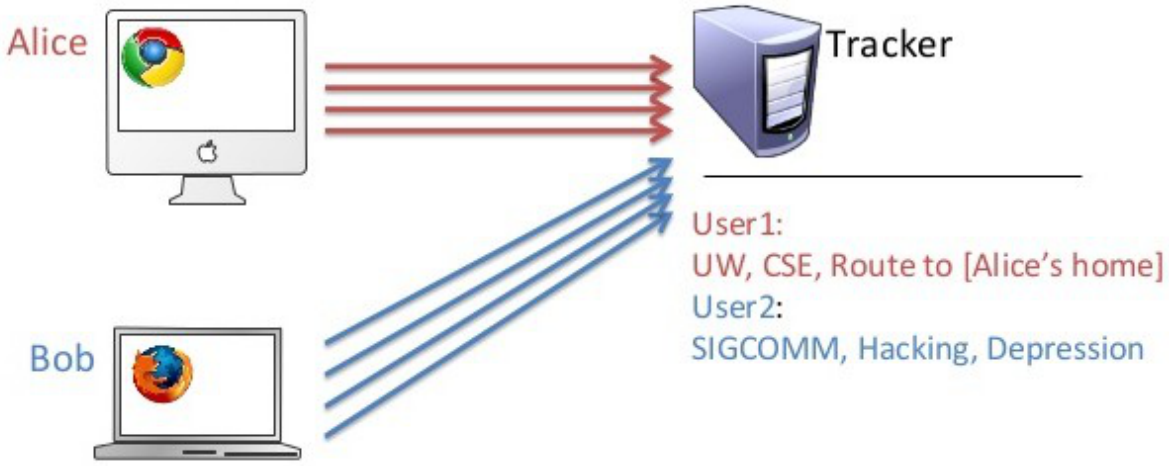
- Network Layer
  - Proxies
  - Tor
  - NAT

Inflexible and some (Tor) are slow. Break applications that rely on IP address for identification

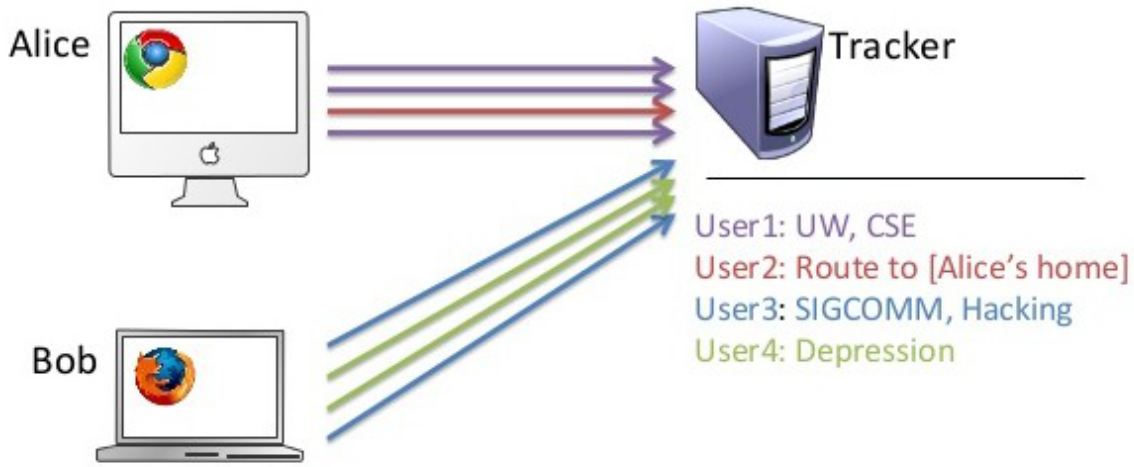
Pseudonyms: power to the user!

- Unlinkable identities that each look like a single user
- Custom policies for how and when to use different pseudonyms
- Necessitates changes at the network and application layer
- (ab)use the massive address space of IPv6 to assign an IP address per pseudonym

Traffic without Pseudonyms



Traffic with Pseudonyms



Contributions

- Pseudonym concept and design
- Case study of one application using pseudonym concept: Web Browser
- Chrome plugin with web gateway as proof of concept

Threat Model

"prevent remote services, with which a user interacts, from linking the user's activities except in ways that the user intends"

- Adversaries want to correlate behavior to link pseudonyms belonging to the same user
- Adversaries can collude with endhosts in the pseudonym system
- Users have to trust first-hop ISPs (or send packets through Tor to someone they trust)

Pseudonyms

- To a user: a collection of activities the user wants linked together
- To a tracker: a single machine

Example Policies

Sort by Identity

Users of the same computer want different identities or want to separate different interests/activities

Users create a different pseudonym for each activity

Example Policies

Banking Websites

Track users to combat fraud, double check when user logs in via a new machine

User creates a pseudonym specifically for banking activities

Example Policies

Separate Sessions

Users don't want requests linked together (e.g., bittorrent downloads)

User creates a pseudonym for each use of the system

For BitTorrent, need browser and BitTorrent client support

Example Policies

Block Third-Party Tracking

Use one identity for requests to a website, but then use random pseudonyms for third party requests

Design

- What needs to be done at each layer?
- What network/OS support is necessary for multiple IPs/machine?
- How are packets sorted into pseudonyms and what are useful policies?

Application Layer

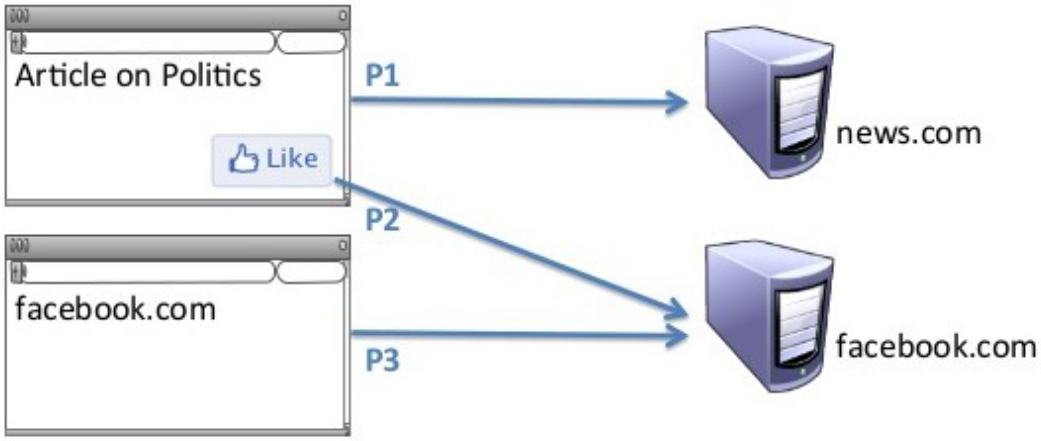
- simple API: allocpseudonym and freepseudonym
- Pseudonym state is application-specific
- Applications decide when to use which pseudonyms

Example: Pseudonymous Web Browser

- Separate cookies, Flash objects, local storage, DNS caches,... for each pseudonym
- Provide default as well as scriptable policies

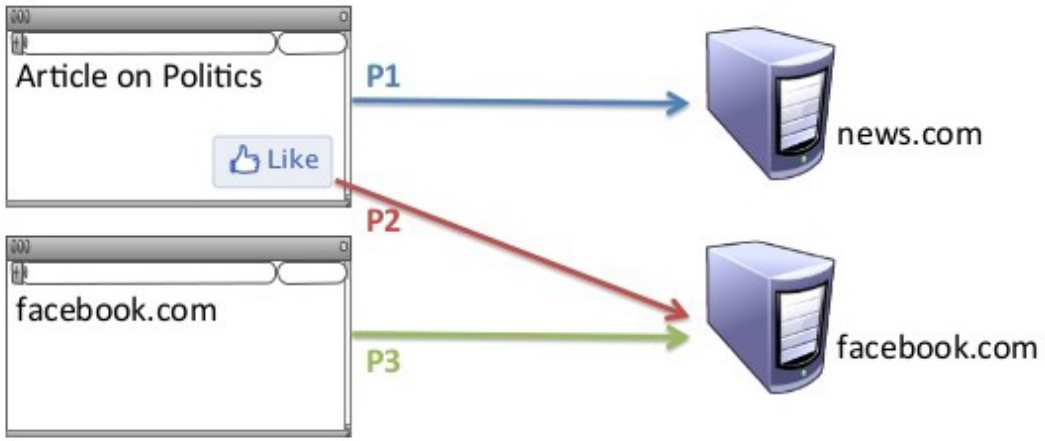
Browser Policies

Default: one pseudonym



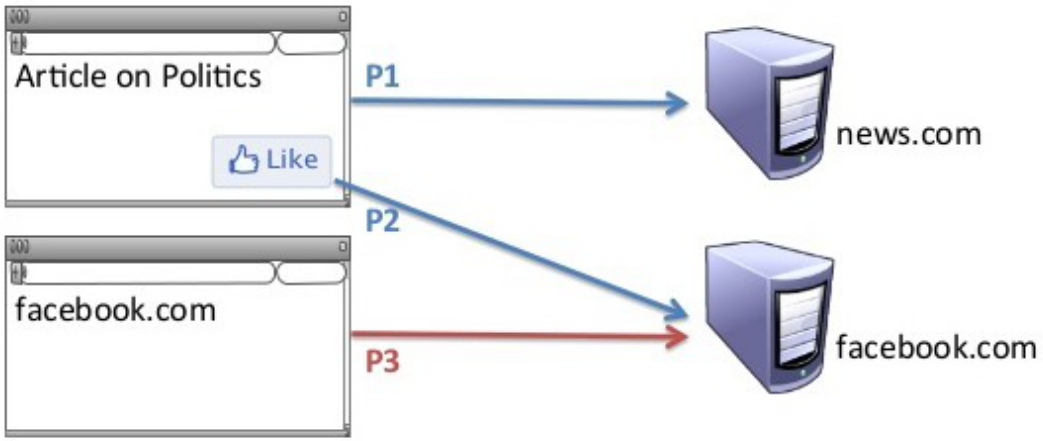
Browser Policies

Per-request: new pseudonym for each request



Browser Policies

Per-1st-Party: new pseudonym for each domain



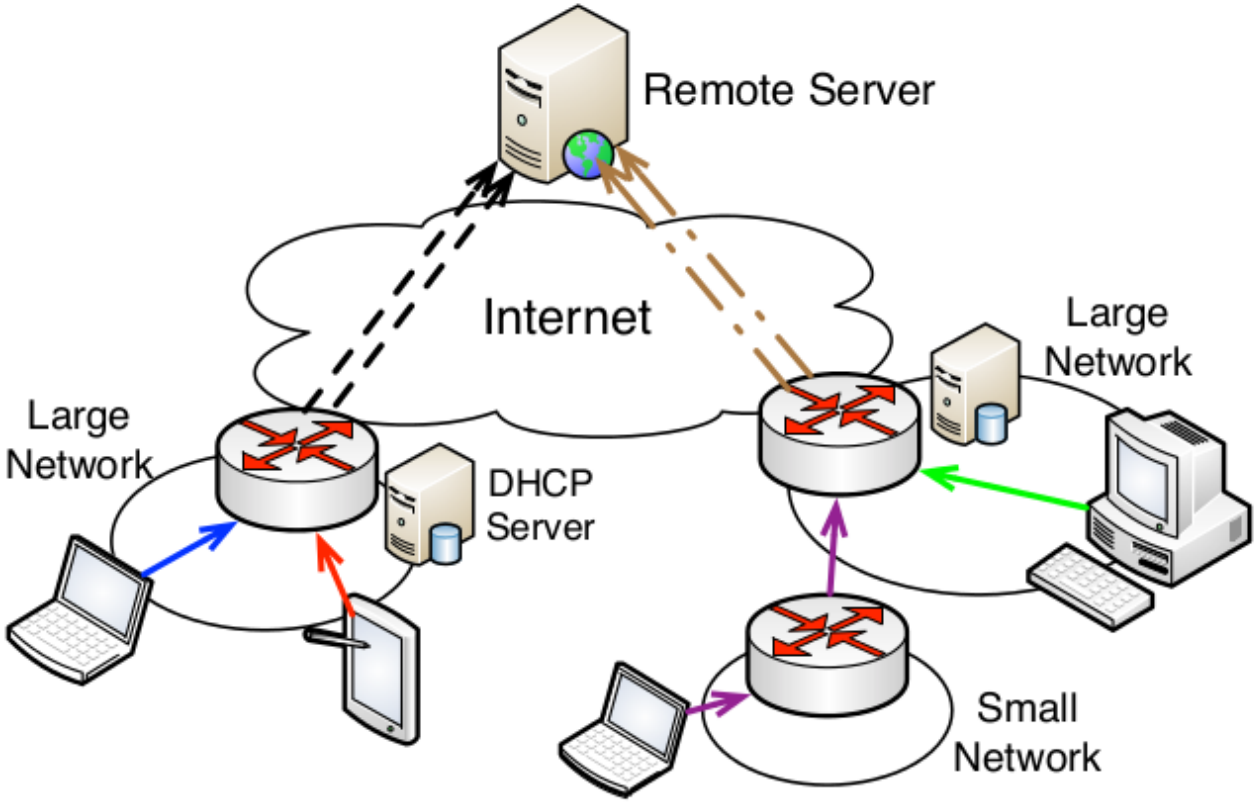
How to allocate Pseudonyms?

- Can't just give each user a bunch of IP addresses if they can just be linked together
- Need a larger pool of IP addresses to choose from and have them randomly assigned (first-hop ISP)

Network Layer Design Goals

- Proper Mixing: pseudonym IPs appear random
- Efficient Routing
- Easy revocation: pseudonym -> IP mapping can be changed efficiently

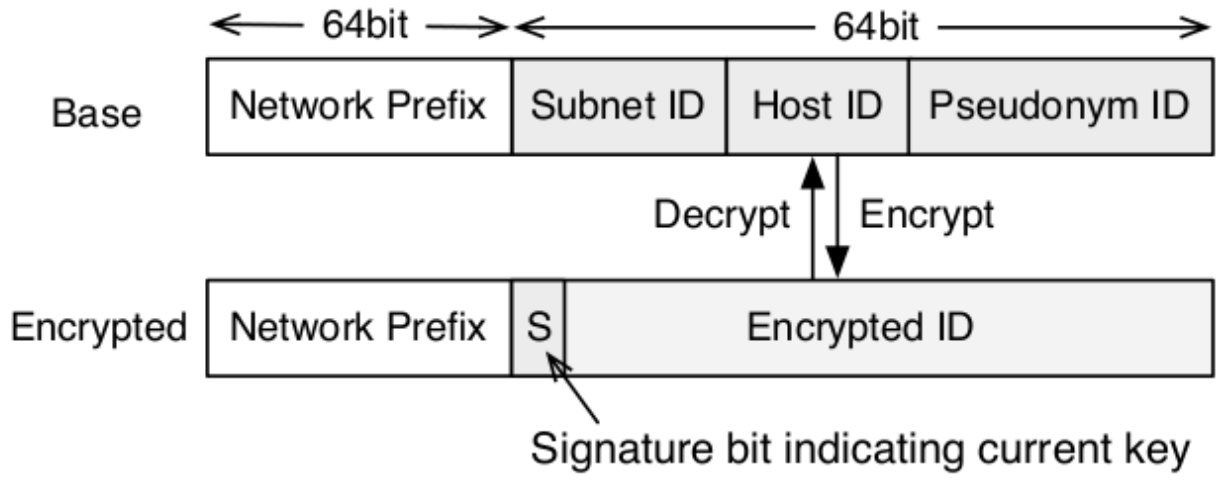
Network Layer



Network Layer

- IPv6 address space is HUGE: 2^128
- Small networks get blocks with a /64 prefix, still 2^64 addresses
- Separate remaining bits into subnet, host and pseudonym ids
- Routers and DHCP servers within the network maintain a secret key for encrypting/decrypting these 3

Network Layer



Proper Mixing

- DHCP allocates new pseudonyms
- Machine broadcasts DHCP request with MAC address and number of desired ids
- Server uses the secret key to encrypt the randomly generated addresses
- Random pseudonym ids create randomly distributed IP addresses after encryption

Efficient Routing

- Reuses network prefix portion of IP address to keep inter-domain routing the same
- Routers decrypt the IP address and then route normally within the ISP's network (don't overwrite the packets)
- No increase in router state!

Easy Revocation

- Routers and DHCP servers keep 2 secret keys at a time
- Only use pseudonyms with a particular signature bit for each key
- New key phases out the old one gradually

Deployability

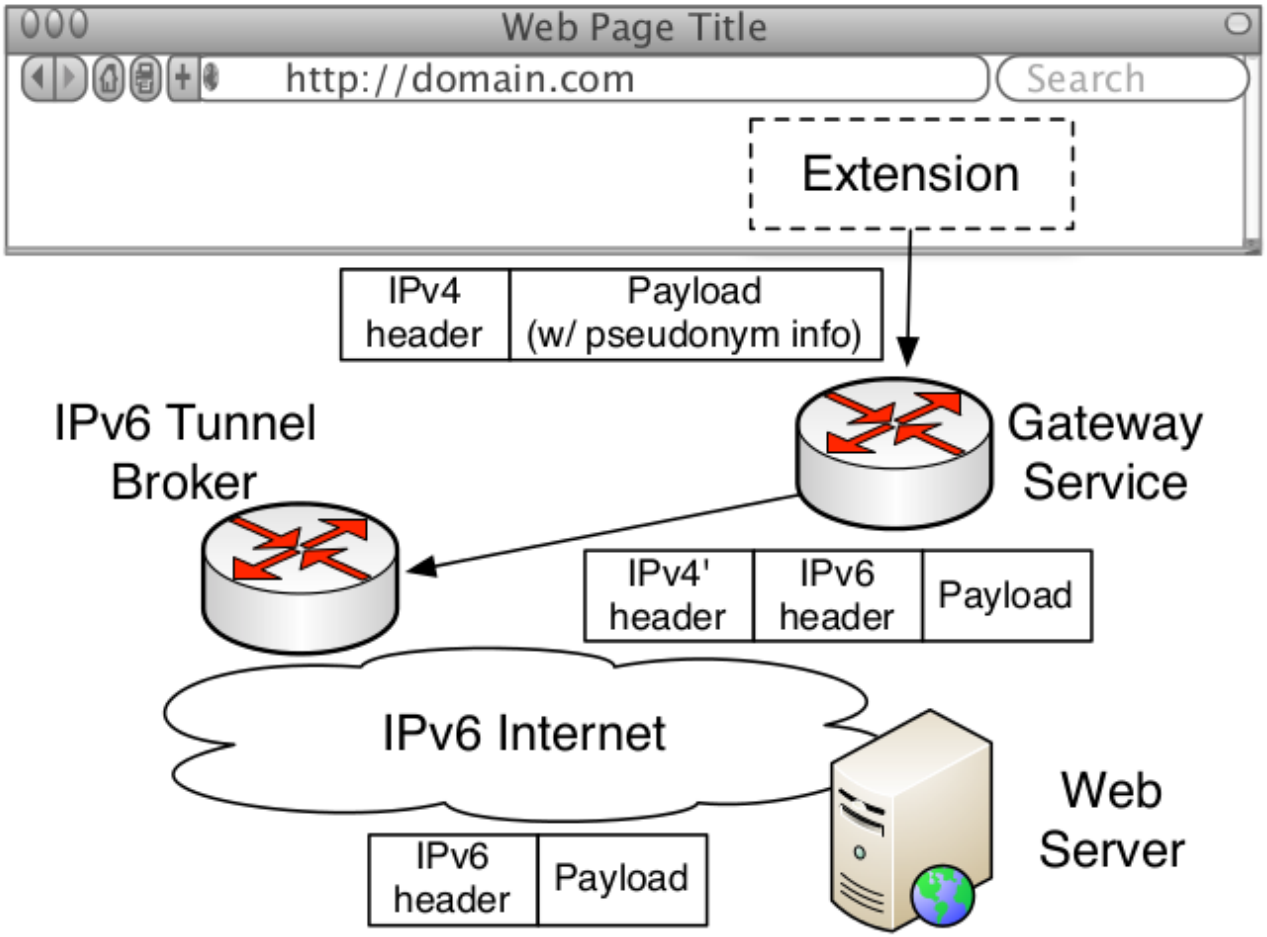
- Requires changes to routers for intra-domain routing
- Requires changes to DHCP servers

Deployability

- Ease adoption by deploying translator routers at the edge to change IPs on the fly
- Use tunneling and proxies to deploy on networks without IPv6

Approximate Implementation

Authors created chrome extension to allow for usage with IPv4 and no ISP support



Pseudonym-specific state

Pseudonyms have their own

- IP Address
- DNS cache
- Cookies/local storage

Leave browser fingerprinting for future work

IPv4

- Only 14.6% of ASes run IPv6
- Tunnel IPv6 through IPv4
- Users have a private IPv4 address used to communicate with gateway
- Users have many public IPv6 pseudonym addresses assigned by the gateway, mimicking the role of the subnet

Evaluation

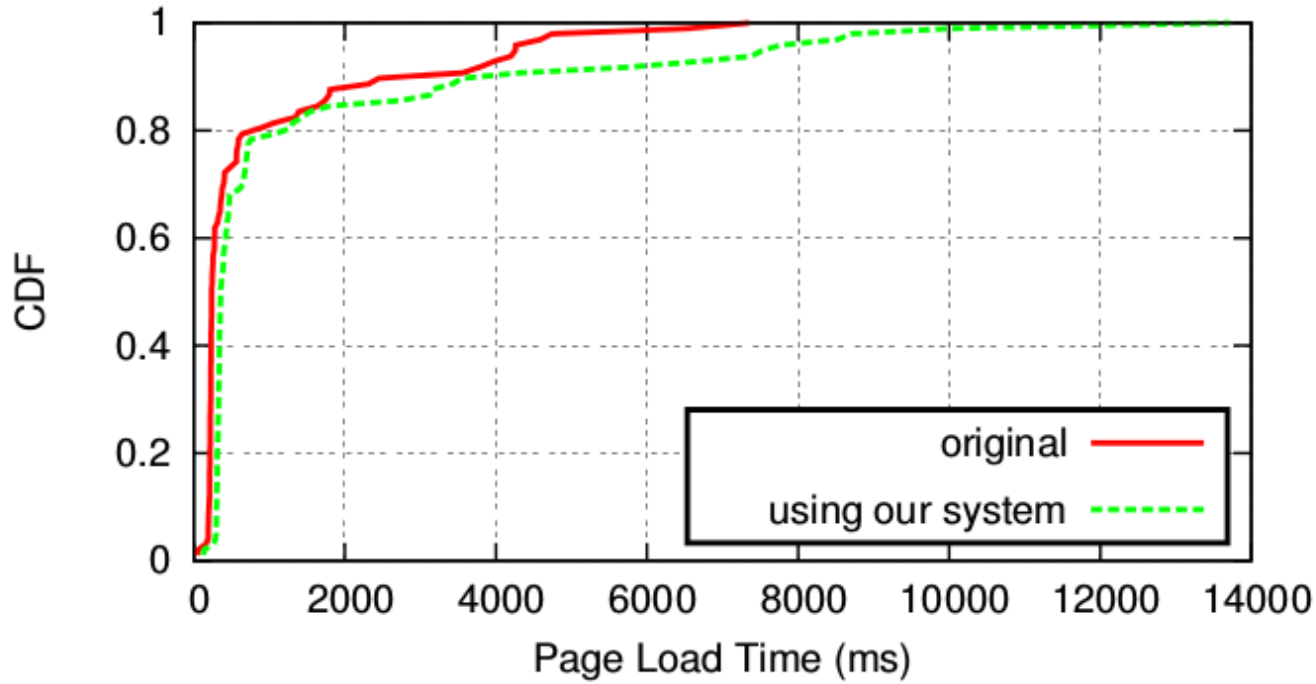
- What's the overhead?
- How expressive is the model
- How many pseudonyms do you need to maintain privacy?

Performance

End-to-end

Top 100 Alexa websites supporting IPv6

- 25% of Alexa Top 100 support IPv6
- 100th IPv6 supporting is ranked 869th overall



Performance

OS

- Mostly negligible slowdown from using hundreds of IP addresses
- Large slowdowns are fixable with better implementation

Performance

Router

- Decryption shown in microbenchmarks to be capable of line-speed.

Expressiveness

Authors implemented various policies

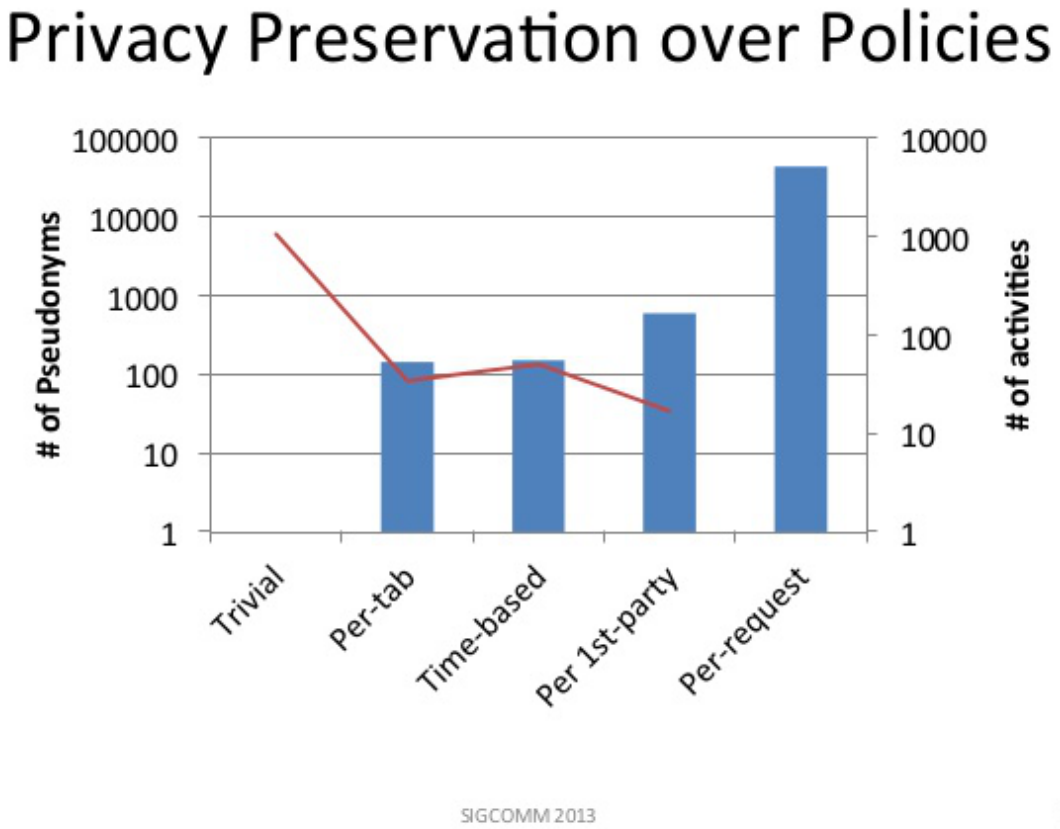
- Per tab
- Per session
- Per 1st-party
- Per page
- 3rd-party blocking
- Per request
- New pseudonym every 10 minutes (Tor)

Privacy Control

- Use HTTP request traces for 3 days of 8 users
- 406 unique domains, 281 (69.2%) containing third-party trackers

Privacy Control

Measure privacy by collusion graphs to see which activities are seen by which parties



Privacy Control

- Need <10 bits for large increase in privacy
- Different browsing patterns benefit from different policies

Conclusion

- Pseudonyms can give us the benefits of trackers without throwing away privacy
- Pseudonyms enable new application layer possibilities
- IPv6's massive address space enables efficient implementation