Vanish: Increasing Data Privacy with Self-Destructing Data

Roxana Geambasu
Yoshi Kohno
Amit Levy
Hank Levy

University of Washington

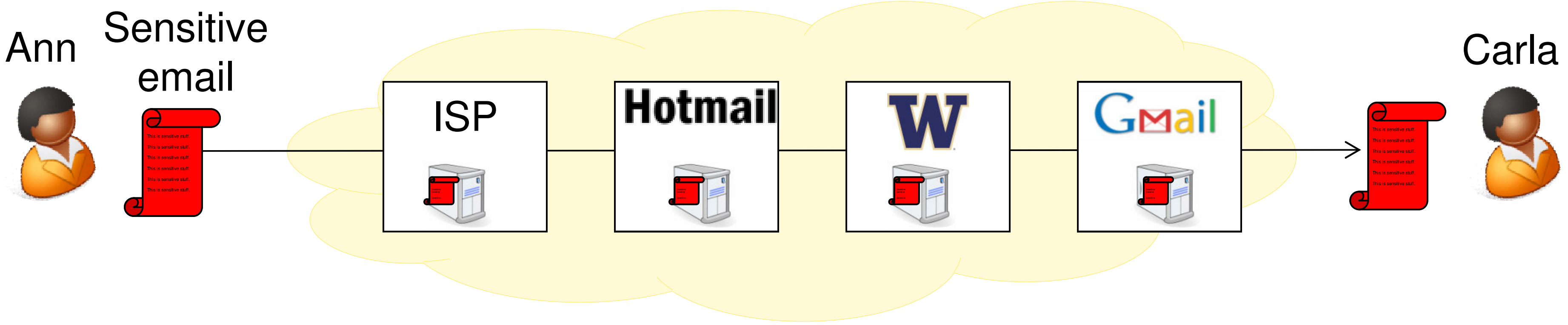
Outline

Part 1: Introducing Self-Destructing Data

Part 2: Vanish Architecture and Implementation

Part 3: Evaluation and Applications

Motivating Problem: Data Lives Forever



Ann

Sensitive email

ISP

Hotmail

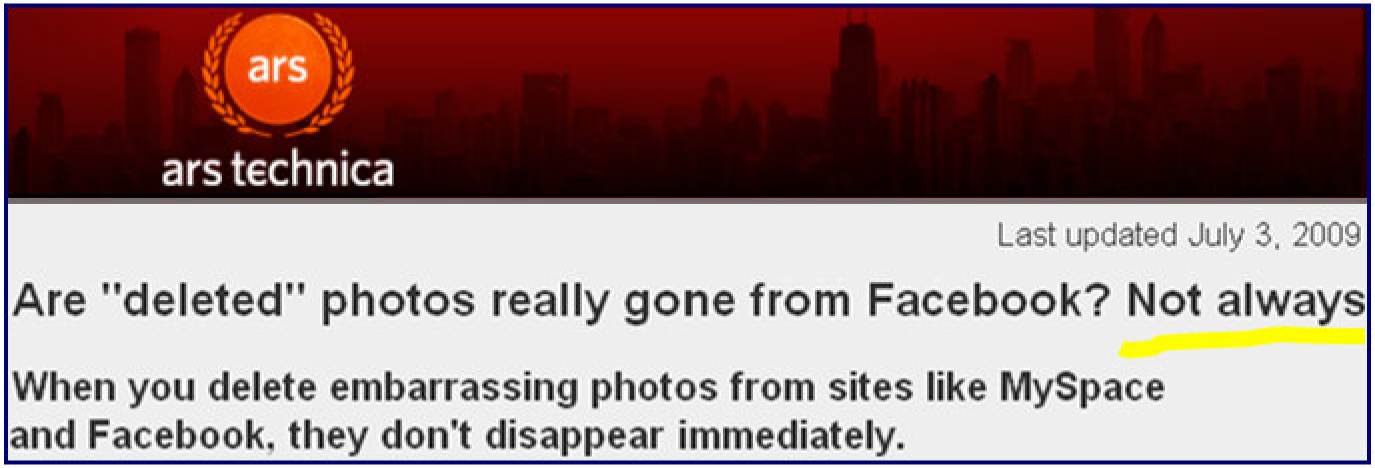
W

Gmail

Carla

How can Ann delete her sensitive email?

- She doesn’t know where all the copies are
- Services may retain data for long after user tries to delete

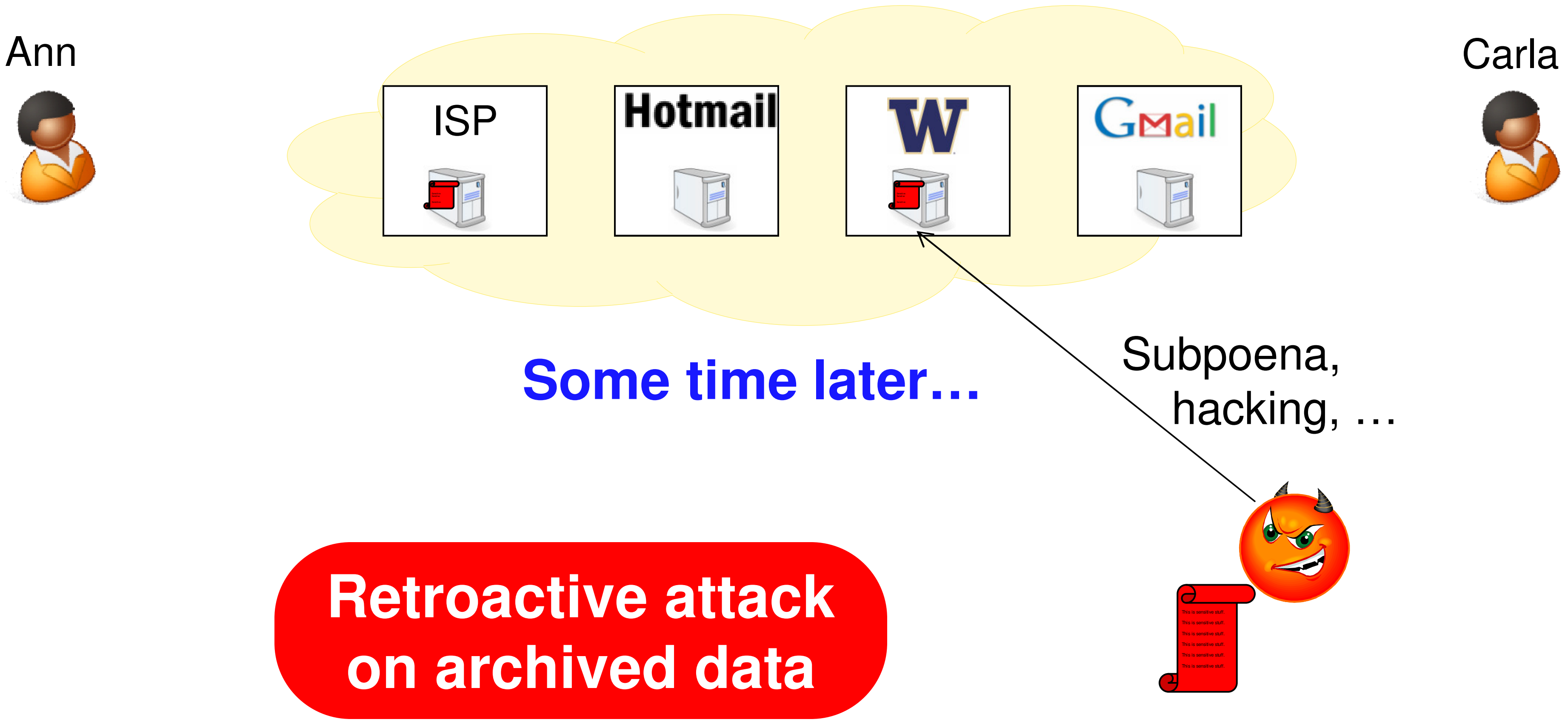


Last updated July 3, 2009

Are "deleted" photos really gone from Facebook? Not always

When you delete embarrassing photos from sites like MySpace and Facebook, they don't disappear immediately.

Archived Copies Can Resurface Years Later

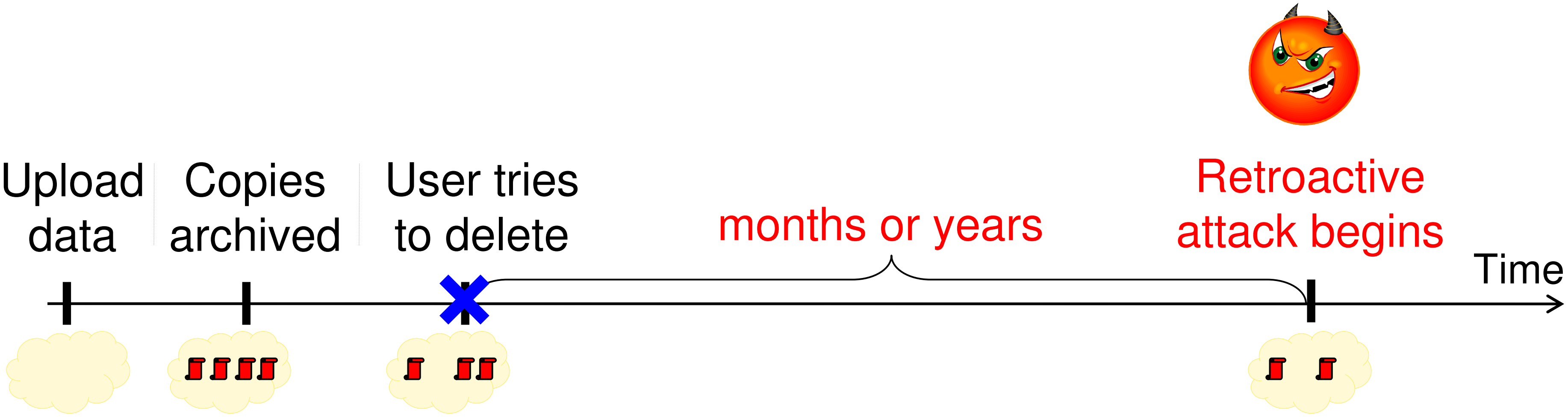


Subpoena, hacking, …

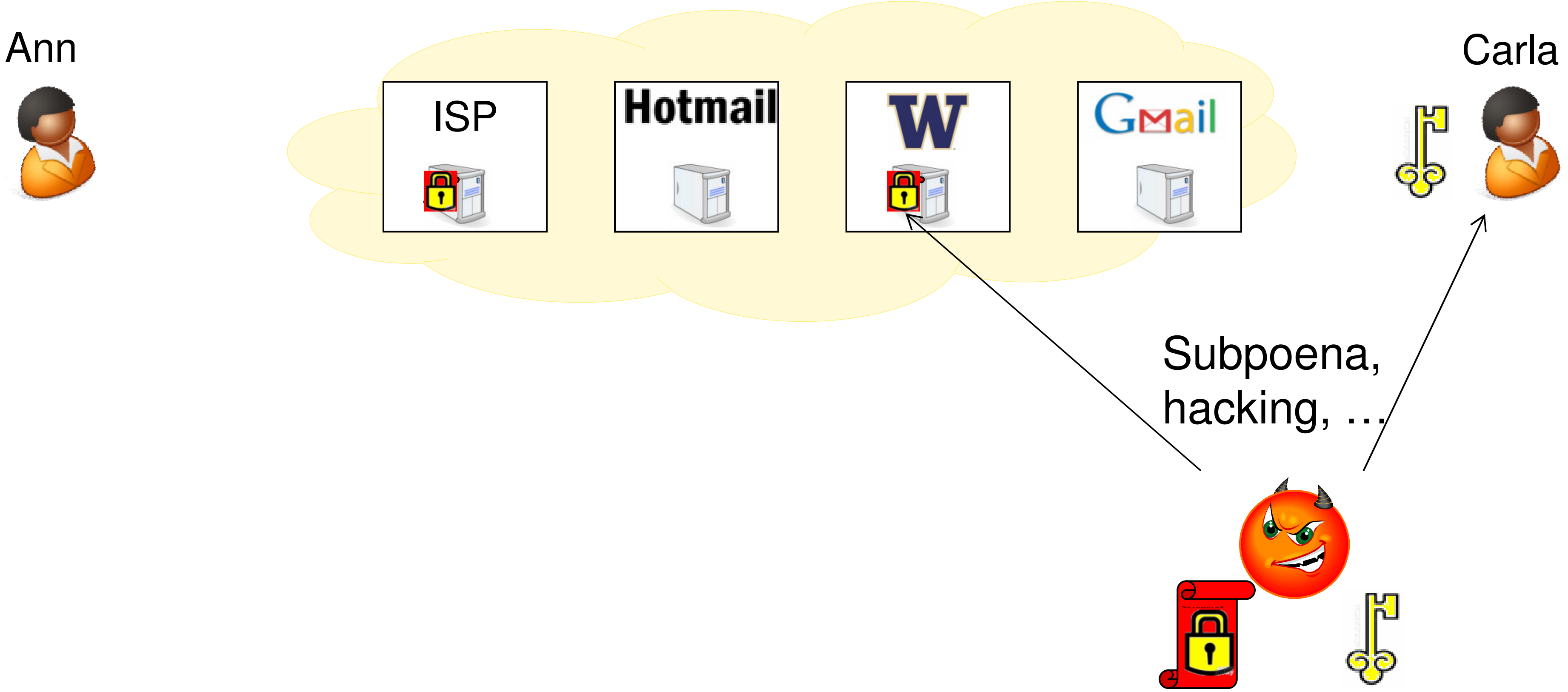
Some time later…

Retroactive attack on archived data

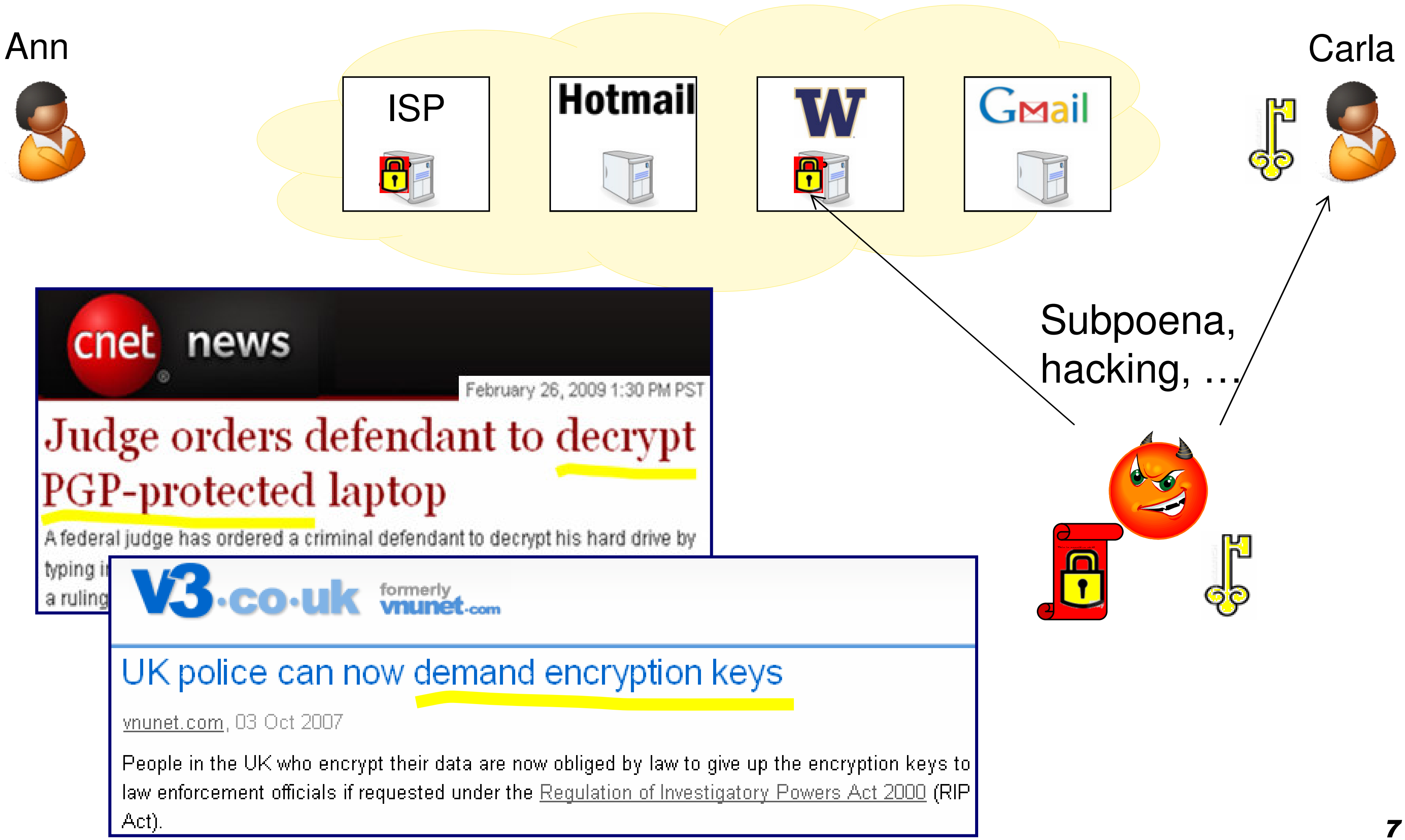
The Retroactive Attack



Why Not Use Encryption (e.g., PGP)?



Why Not Use Encryption (e.g., PGP)?



cnet news

February 26, 2009 1:30 PM PST

Judge orders defendant to decrypt PGP-protected laptop

A federal judge has ordered a criminal defendant to decrypt his hard drive by typing i

a ruling

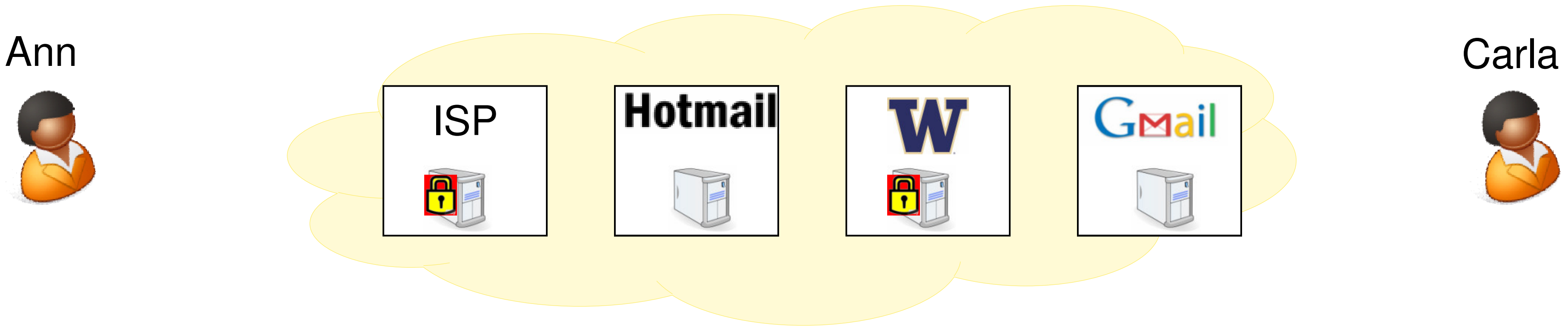
V3.co.uk formerly vnunet.com

UK police can now demand encryption keys

vnunet.com, 03 Oct 2007

People in the UK who encrypt their data are now obliged by law to give up the encryption keys to law enforcement officials if requested under the Regulation of Investigatory Powers Act 2000 (RIP Act).

Why Not Use a Centralized Service?



Ann

ISP

Hotmail

W

Gmail

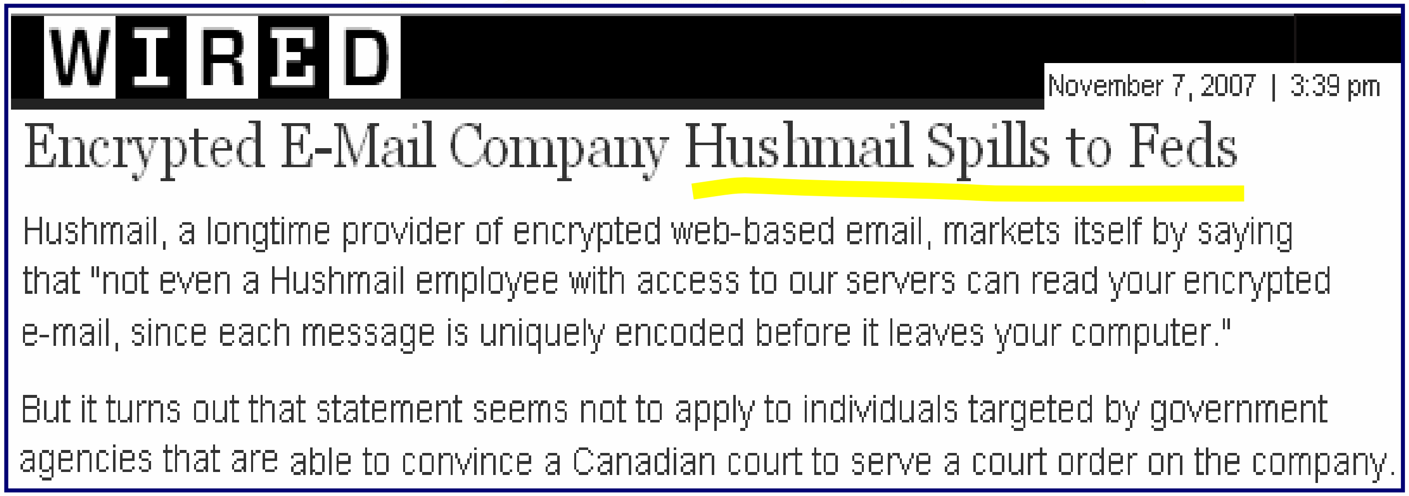
Carla

Centralized Service

"Trust us: we'll help you delete your data on time."

Backdoor agreement

Why Not Use a Centralized Service?



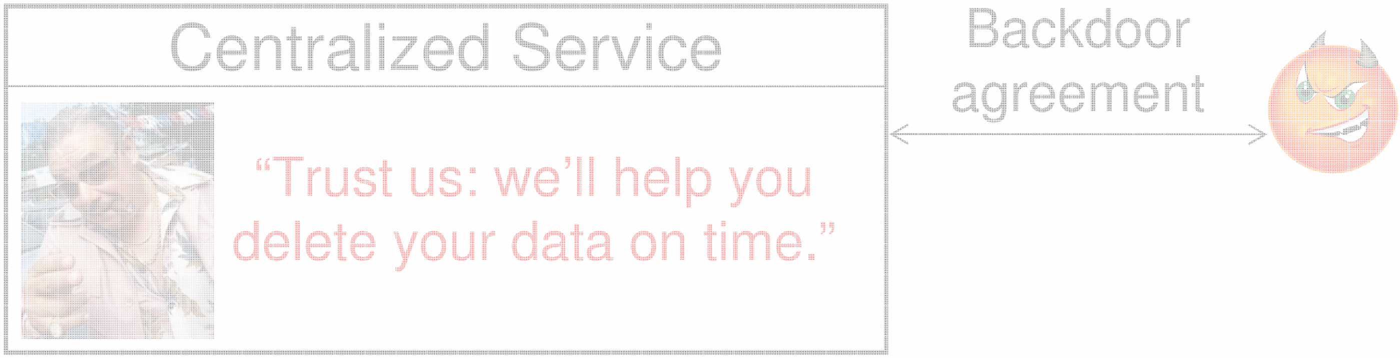
WIRED

November 7, 2007 | 3:39 pm

Encrypted E-Mail Company Hushmail Spills to Feds

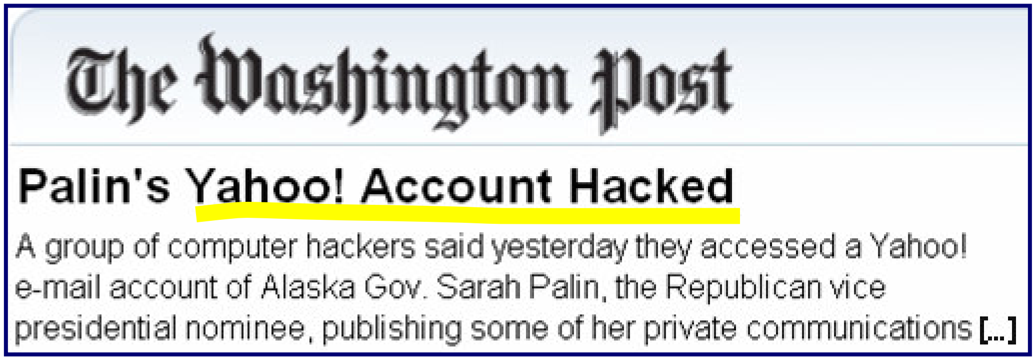
Hushmail, a longtime provider of encrypted web-based email, markets itself by saying that "not even a Hushmail employee with access to our servers can read your encrypted e-mail, since each message is uniquely encoded before it leaves your computer."

But it turns out that statement seems not to apply to individuals targeted by government agencies that are able to convince a Canadian court to serve a court order on the company.



The Problem: Two Huge Challenges for Privacy

1. Data lives forever
   - On the web: emails, Facebook photos, Google Docs, blogs, …
   - In the home: disks are cheap, so no need to ever delete data
   - In your pocket: phones and USB sticks have GBs of storage
2. Retroactive disclosure of both data and user keys has become commonplace
   - Hackers
   - Misconfigurations
   - Legal actions
   - Border seizing
   - Theft
   - Carelessness



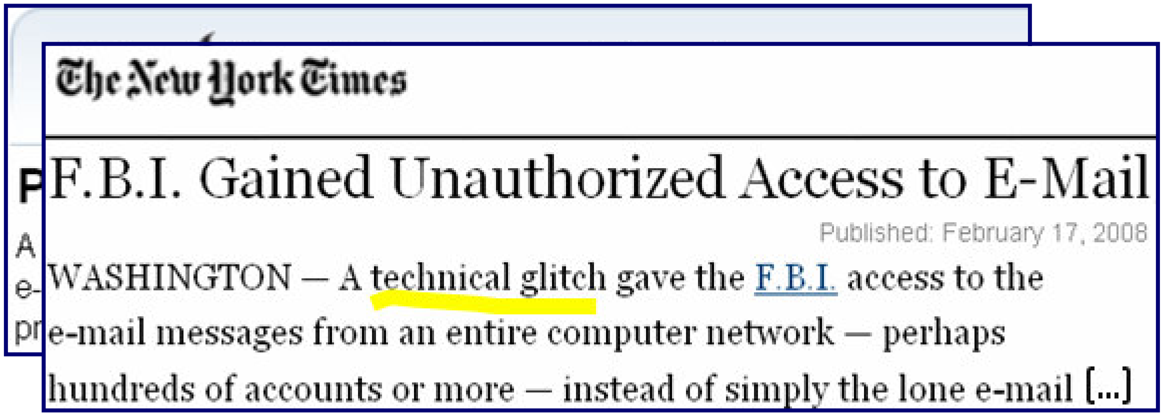
The Washington Post

Palin's Yahoo! Account Hacked

A group of computer hackers said yesterday they accessed a Yahoo! e-mail account of Alaska Gov. Sarah Palin, the Republican vice presidential nominee, publishing some of her private communications [...]

The Problem: Two Huge Challenges for Privacy

1. Data lives forever
   - On the web: emails, Facebook photos, Google Docs, blogs, …
   - In the home: disks are cheap, so no need to ever delete data
   - In your pocket: phones and USB sticks have GBs of storage
2. Retroactive disclosure of both data and user keys has become commonplace
   - Hackers
   - Misconfigurations
   - Legal actions
   - Border seizing
   - Theft
   - Carelessness



The New York Times

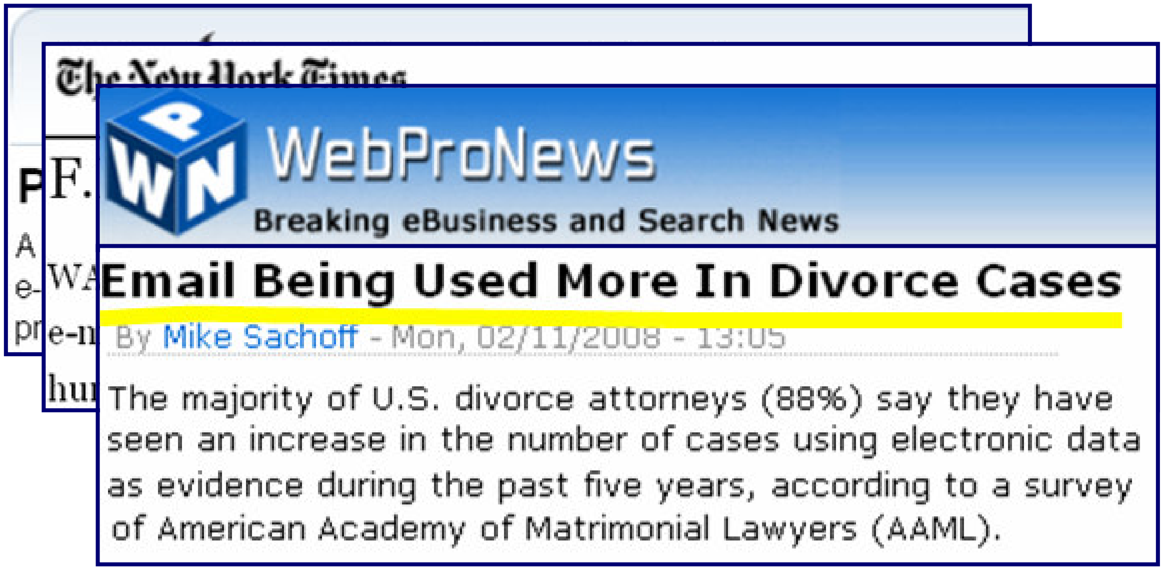
F.B.I. Gained Unauthorized Access to E-Mail

Published: February 17, 2008

WASHINGTON — A technical glitch gave the F.B.I. access to the e-mail messages from an entire computer network — perhaps hundreds of accounts or more — instead of simply the lone e-mail [...]

The Problem: Two Huge Challenges for Privacy

1. Data lives forever
   - □ On the web: emails, Facebook photos, Google Docs, blogs, …
   - □ In the home: disks are cheap, so no need to ever delete data
   - □ In your pocket: phones and USB sticks have GBs of storage
2. Retroactive disclosure of both data and user keys has become commonplace
   - □ Hackers
   - □ Misconfigurations
   - □ Legal actions
   - □ Border seizing
   - □ Theft
   - □ Carelessness



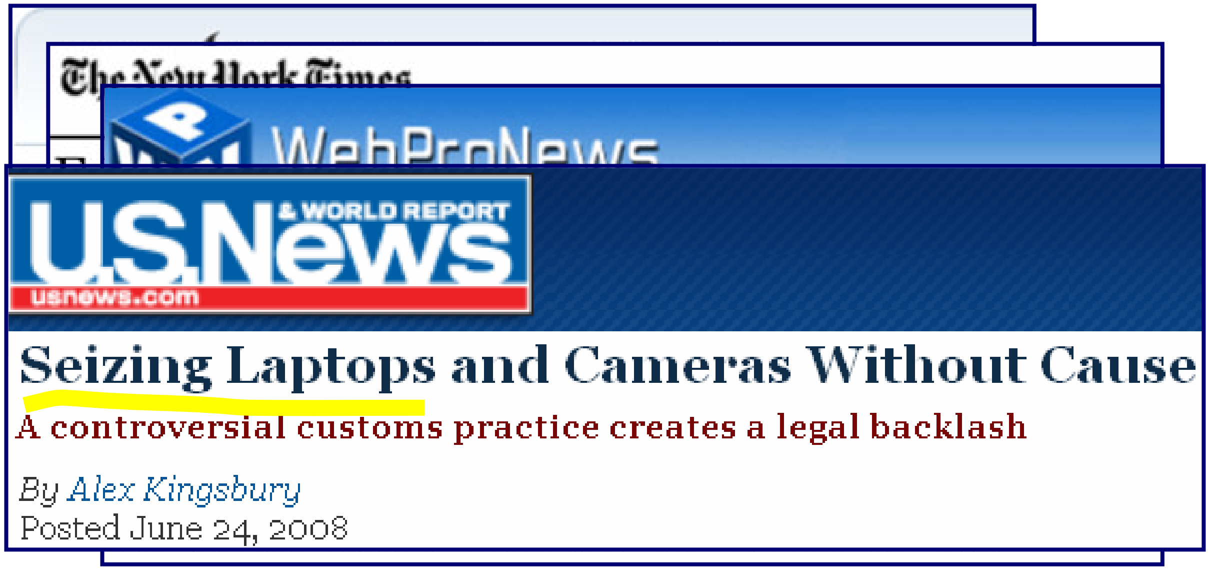
Email Being Used More In Divorce Cases

By Mike Sachoff - Mon, 02/11/2008 - 13:05

The majority of U.S. divorce attorneys (88%) say they have seen an increase in the number of cases using electronic data as evidence during the past five years, according to a survey of American Academy of Matrimonial Lawyers (AAML).

The Problem: Two Huge Challenges for Privacy

1. Data lives forever
   - On the web: emails, Facebook photos, Google Docs, blogs, …
   - In the home: disks are cheap, so no need to ever delete data
   - In your pocket: phones and USB sticks have GBs of storage
2. Retroactive disclosure of both data and user keys has become commonplace
   - Hackers
   - Misconfigurations
   - Legal actions
   - Border seizing
   - Theft
   - Carelessness



Seizing Laptops and Cameras Without Cause

A controversial customs practice creates a legal backlash

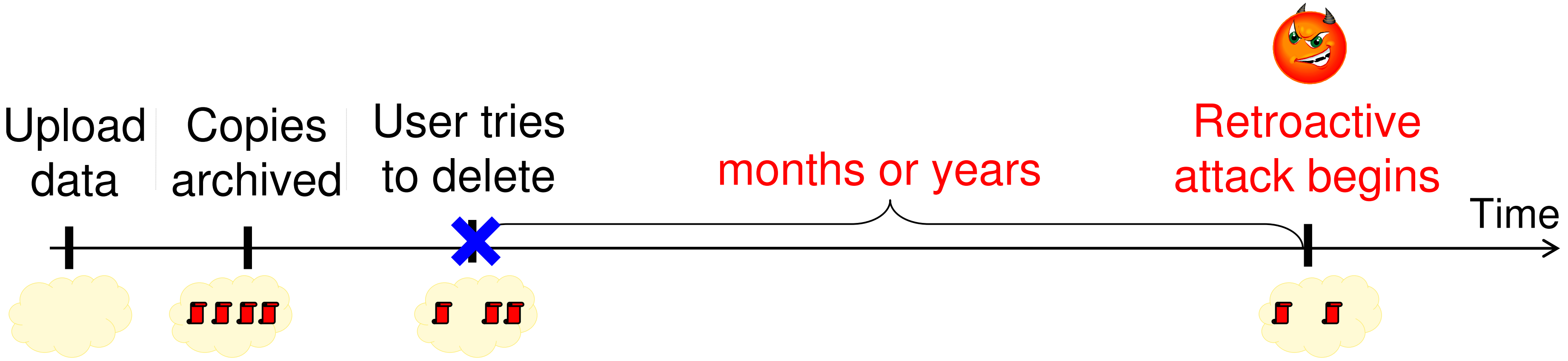
By Alex Kingsbury
Posted June 24, 2008

Question:

Can we empower users with control of data lifetime?

Answer:

Self-destructing data

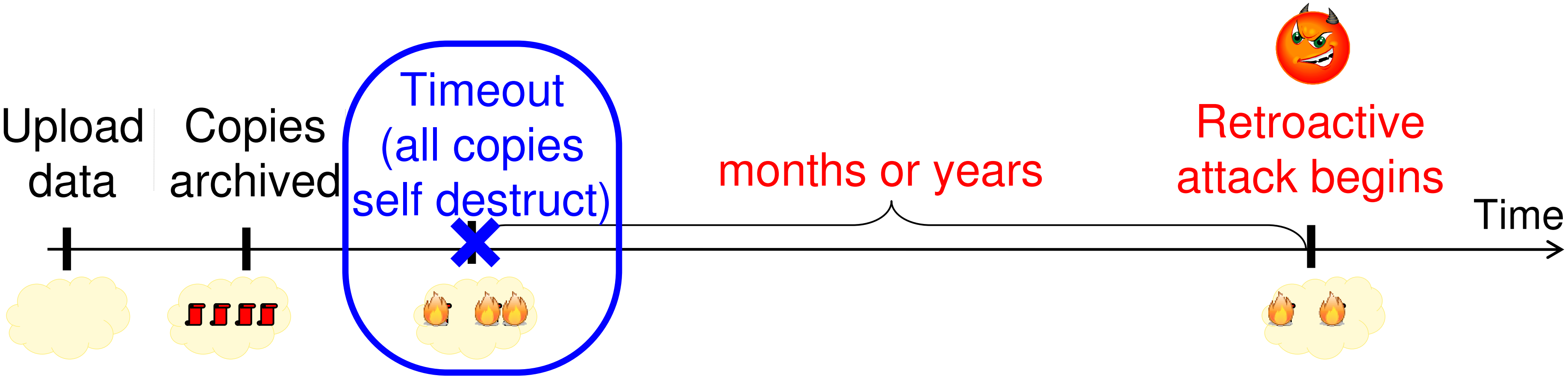


Question:

Can we empower users with control of data lifetime?

Answer:

Self-destructing data



Upload data

Copies archived

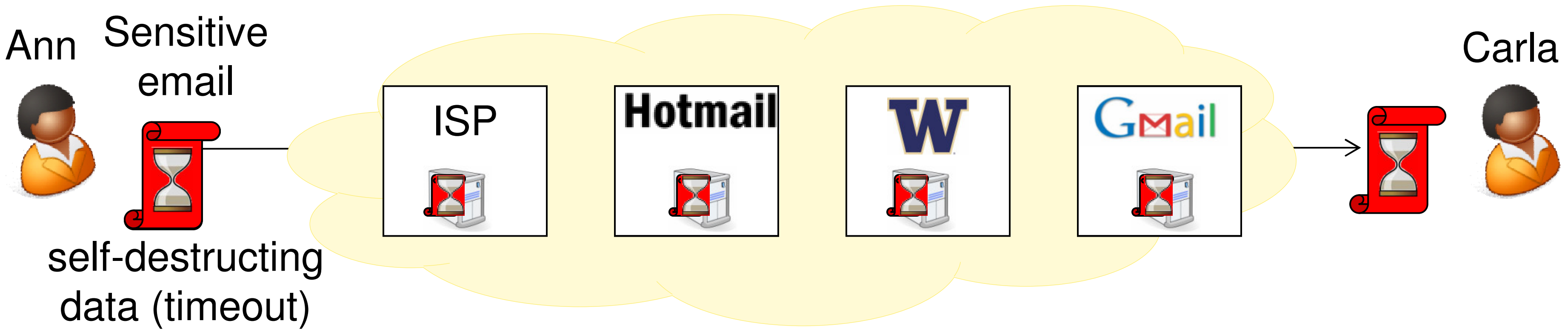
Timeout (all copies self destruct)

months or years

Retroactive attack begins

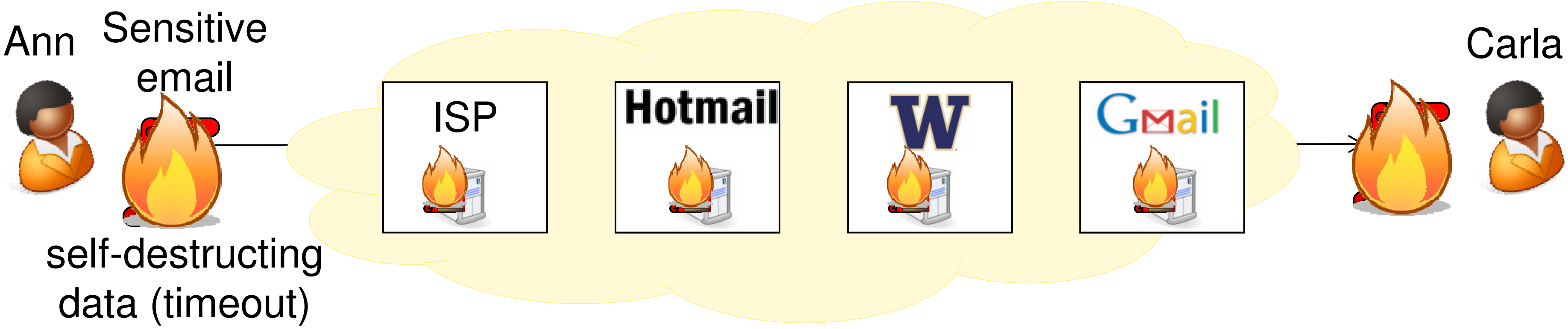
Time

Self-Destructing Data Model



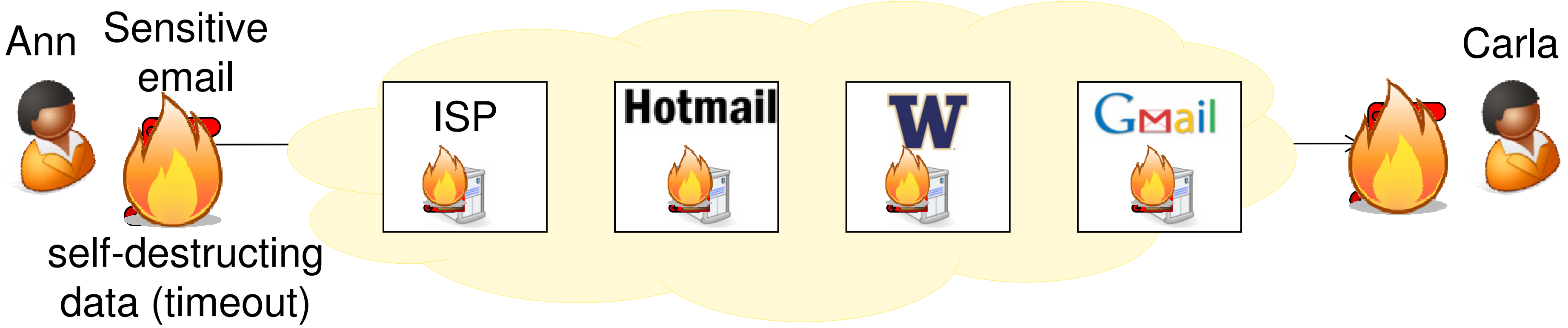
1. Until timeout, users can read original message

Self-Destructing Data Model



1. Until timeout, users can read original message
2. After timeout, all copies become permanently unreadable
   - 2.1. even for attackers who obtain an archived copy & user keys
   - 2.2. without requiring explicit delete action by user/services
   - 2.3. without having to trust any centralized services

Self-Destructing Data Model



Ann

Sensitive email

self-destructing data (timeout)

ISP

Carla

Goals of Self-Destructing Data

1. Until timeout, users can read original message
2. After timeout, all copies become permanently unreadable
   2.1. even for attackers who obtain an archived copy & user keys
   2.2. without requiring explicit delete action by user/services
   2.3. without having to trust any centralized services

Outline

Part 1: Introducing Self-Destructing Data

Part 2: Vanish Architecture and Implementation

Part 3: Evaluation and Applications

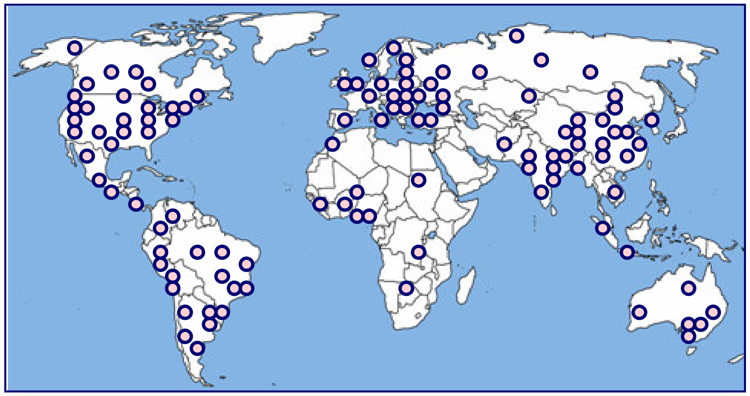
Vanish: Self-Destructing Data System

- Traditional solutions are not sufficient for self-destructing data goals:
  - PGP
  - Centralized data management services
  - Forward-secure encryption
  - …
- Let’s try something completely new!

Idea:
Leverage P2P systems

P2P 101: Intro to Peer-To-Peer Systems

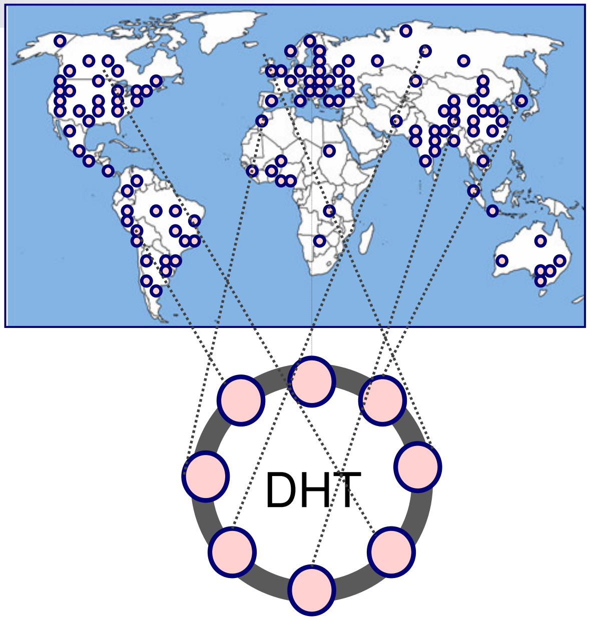
- A system composed of individually-owned computers that make a portion of their resources available directly to their peers without intermediary managed hosts or servers. [~wikipedia]



Important P2P properties (for Vanish):

- Huge scale – millions of nodes
- Geographic distribution – hundreds of countries
- Decentralization – individually-owned, no single point of trust
- Constant evolution – nodes constantly join and leave

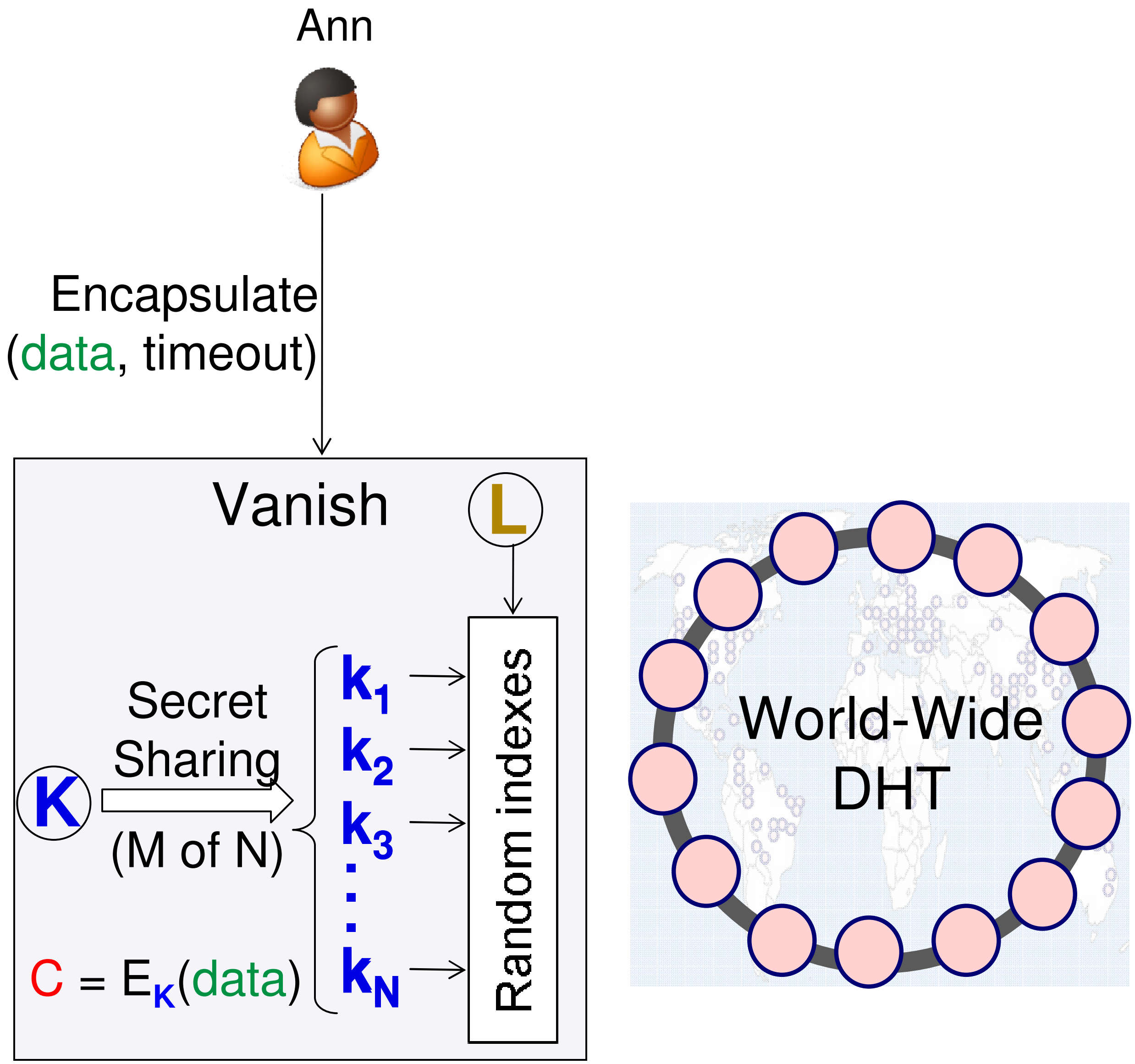
Distributed Hashtables (DHTs)



Logical structure

- Hashtable data structure implemented on a P2P network
  - Get and put (index, value) pairs
  - Each node stores part of the index space
- DHTs are part of many file sharing systems:
  - Vuze, Mainline, KAD
  - Vuze has ~1.5M simultaneous nodes in ~190 countries
- Vanish leverages DHTs to provide self-destructing data
  - One of few applications of DHTs outside of file sharing

How Vanish Works: Data Encapsulation



Ann

Encapsulate (data, timeout)

Vanish

Secret Sharing (M of N)

K → k_1, k_2, k_3, …, k_N

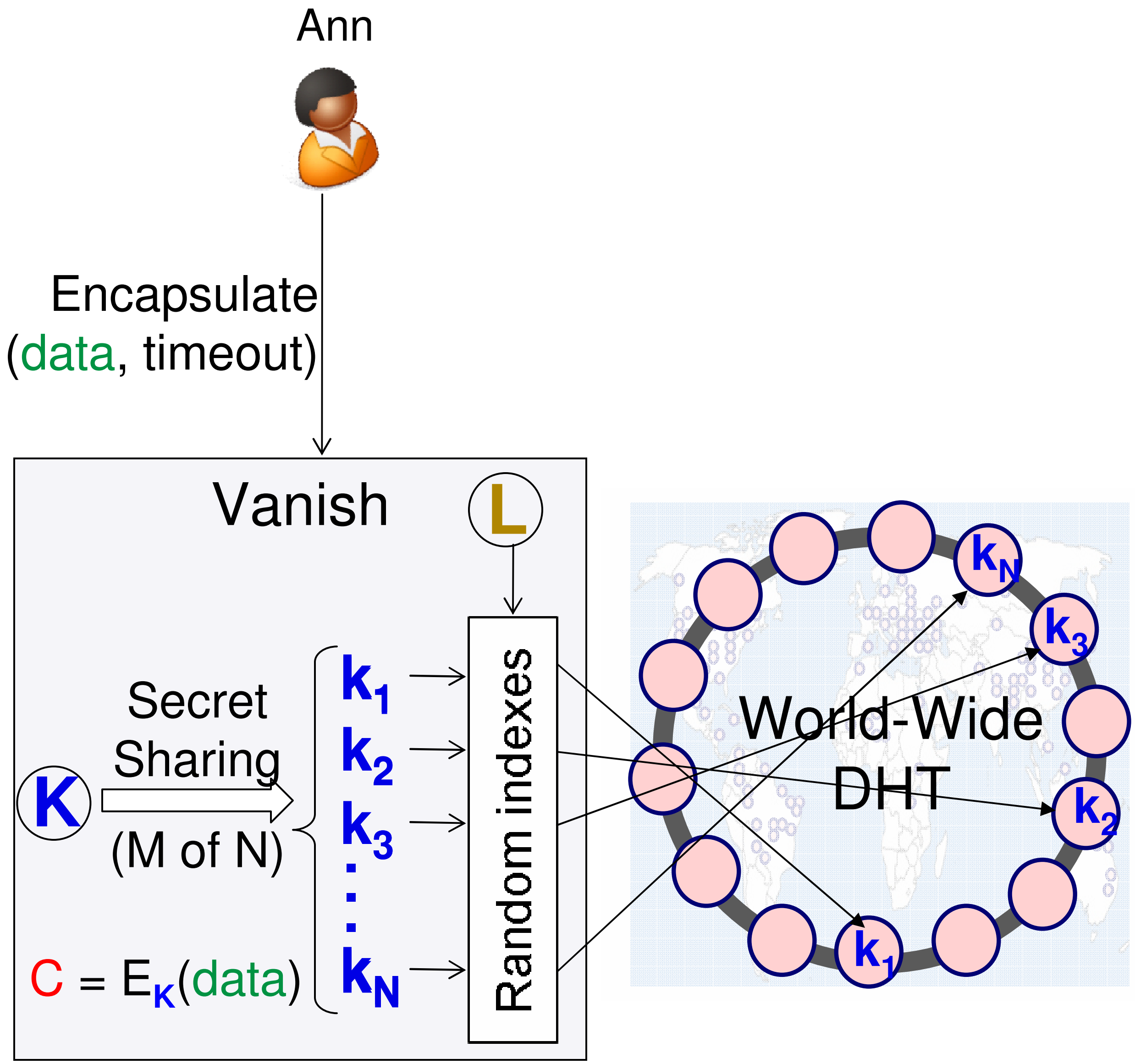
L

Random indexes

$C = E_K(\text{data})$

World-Wide DHT

How Vanish Works: Data Encapsulation



Ann

Encapsulate
(data, timeout)

Vanish

L

K

Secret
Sharing

(M of N)

k_1
k_2
k_3
.
.
.
k_N

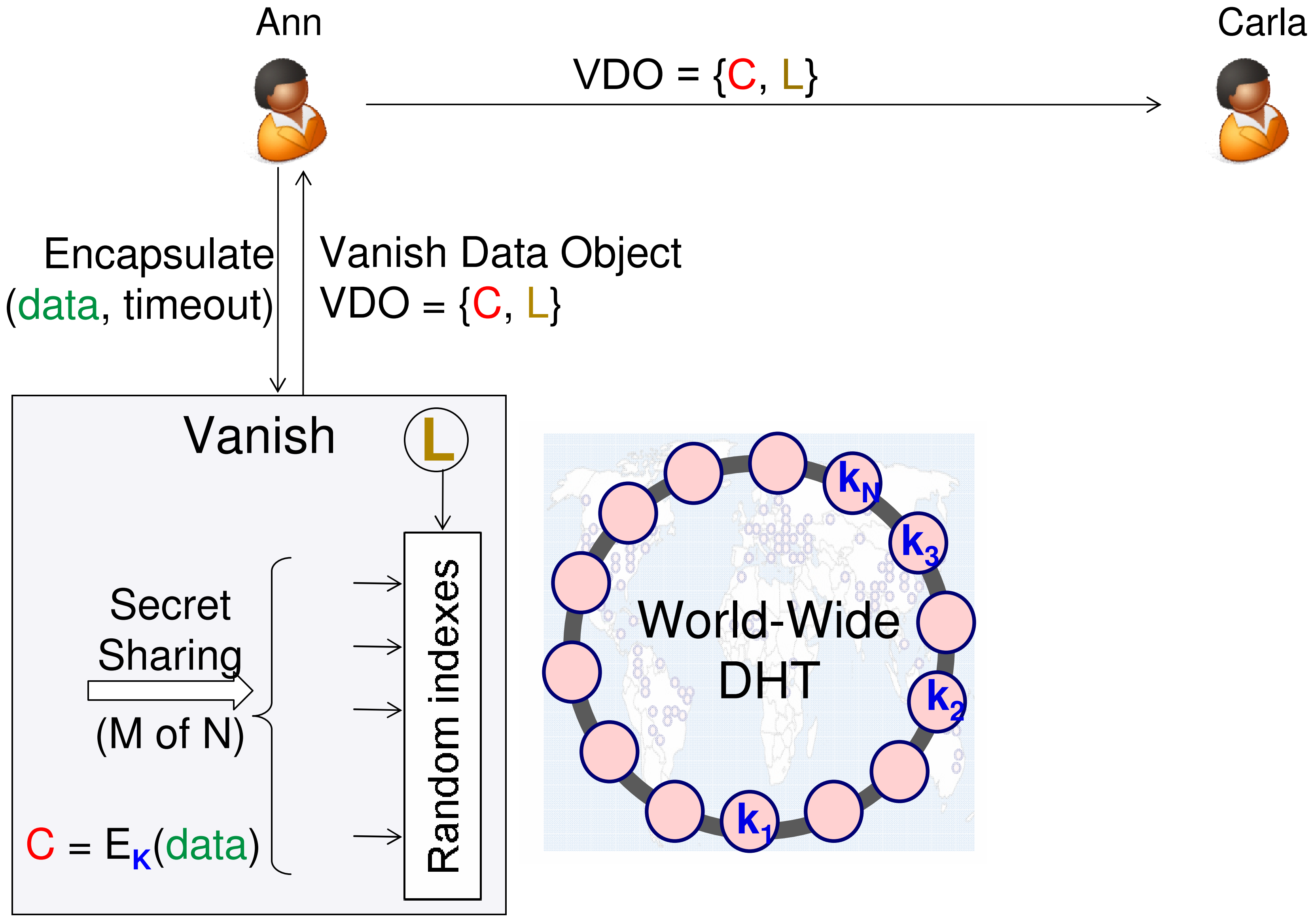
Random indexes

C = E_K(data)

World-Wide
DHT

k_N
k_3
k_2
k_1

How Vanish Works: Data Encapsulation



Ann

Carla

VDO = {C, L}

Encapsulate
(data, timeout)

Vanish Data Object
VDO = {C, L}

Vanish

L

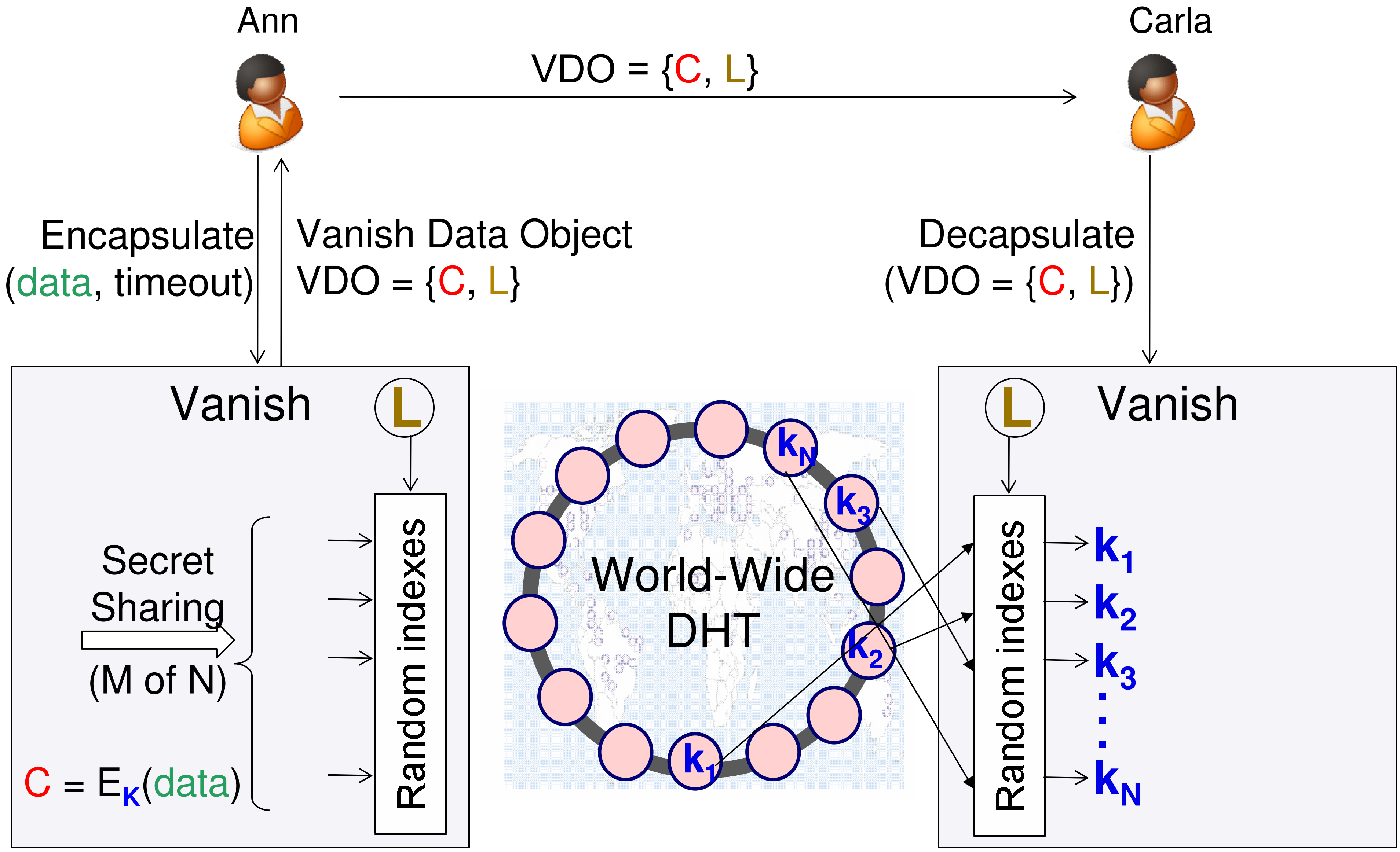
Secret
Sharing
(M of N)

$C = E_K(\text{data})$

Random indexes

World-Wide
DHT

How Vanish Works: Data Decapsulation



Ann

Carla

VDO = {C, L}

Encapsulate (data, timeout)

Vanish Data Object VDO = {C, L}

Decapsulate (VDO = {C, L})

Vanish

L

Secret Sharing (M of N)

$C = E_K(\text{data})$

Random indexes

World-Wide DHT

k_N

k_3

k_2

k_1

Vanish

L

Random indexes

k_1

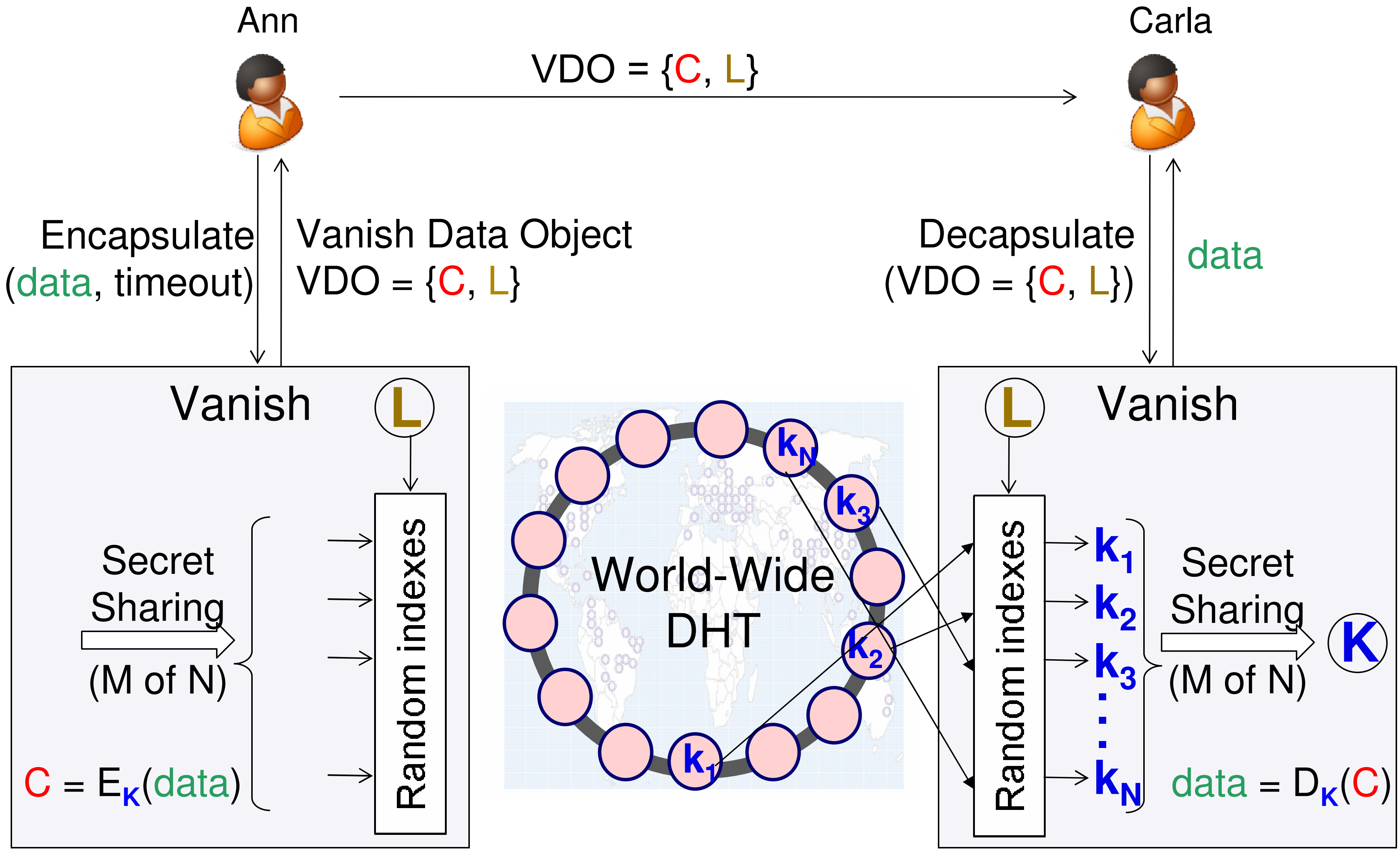
k_2

k_3

:

k_N

How Vanish Works: Data Decapsulation



Ann

Carla

VDO = {C, L}

Encapsulate (data, timeout)

Vanish Data Object VDO = {C, L}

Decapsulate (VDO = {C, L})

data

Vanish

L

Secret Sharing (M of N)

$C = E_K(\text{data})$

Random indexes

World-Wide DHT

k_N

k_3

k_2

k_1

Vanish

L

Random indexes

k_1

k_2

k_3

$\vdots$

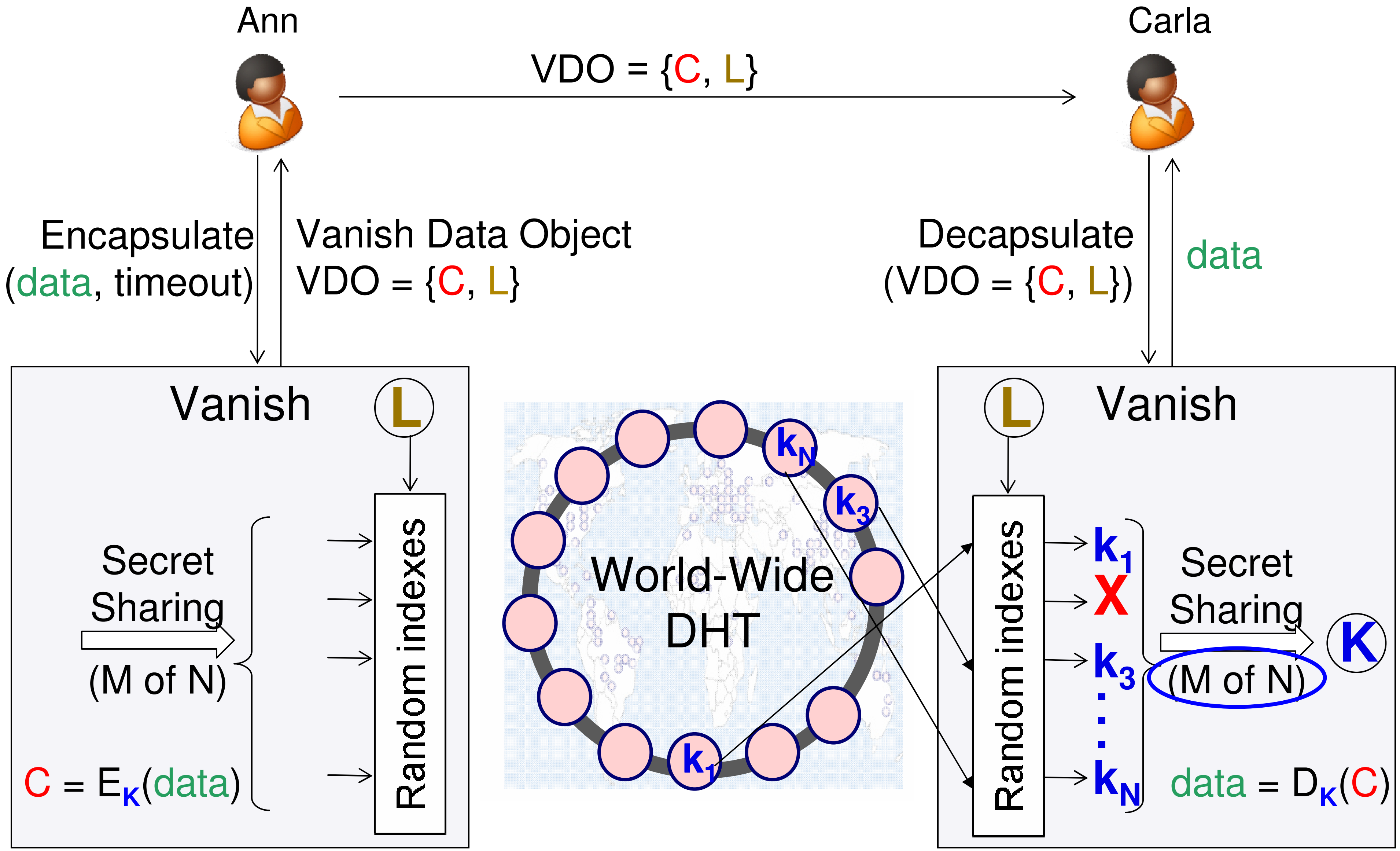
k_N

Secret Sharing (M of N)

K

$\text{data} = D_K(C)$

How Vanish Works: Data Decapsulation



Ann

Carla

VDO = {C, L}

Encapsulate (data, timeout)

Vanish Data Object VDO = {C, L}

Decapsulate (VDO = {C, L})

data

Vanish

L

Secret Sharing

(M of N)

$C = E_K(\text{data})$

Random indexes

World-Wide DHT

k_N

k_3

k_1

Vanish

L

Random indexes

k_1

X

k_3

⋮

k_N

Secret Sharing

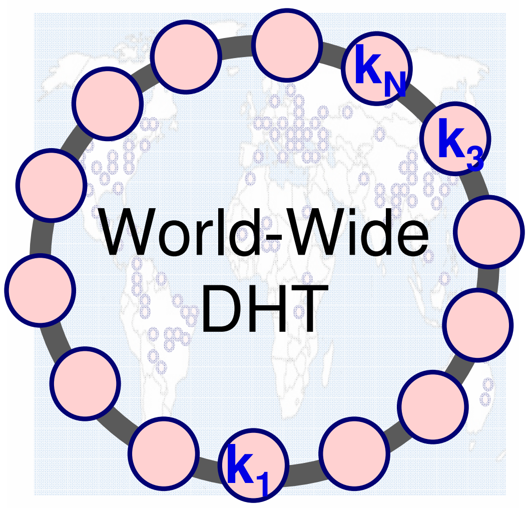
(M of N)

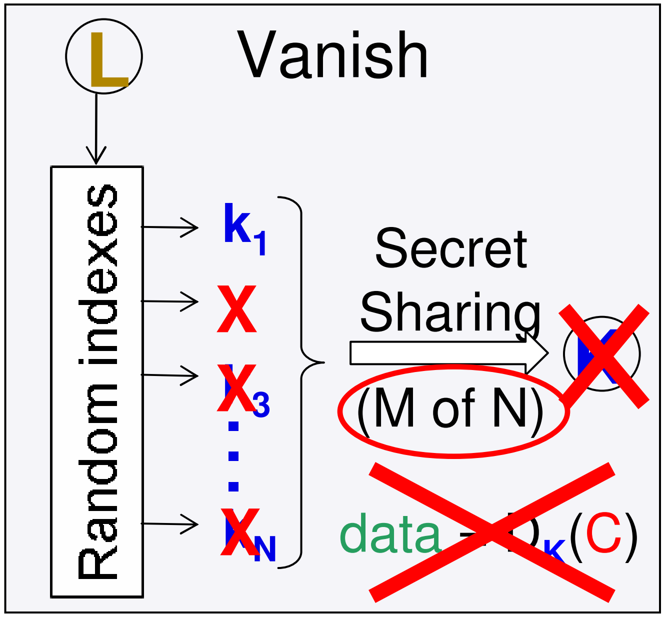
K

$\text{data} = D_K(C)$

How Vanish Works: Data Timeout

- The DHT loses key pieces over time
  - □ Natural churn: nodes crash or leave the DHT
  - □ Built-in timeout: DHT nodes purge data periodically





- Key loss makes all data copies permanently unreadable

Outline

Part 1: Introducing Self-Destructing Data

Part 2: Vanish Architecture and Implementation

Part 3: Evaluation and Applications

Evaluation

- Experiments to understand and improve:
  1. data availability before timeout
  2. data unavailability after timeout
  3. performance
  4. security

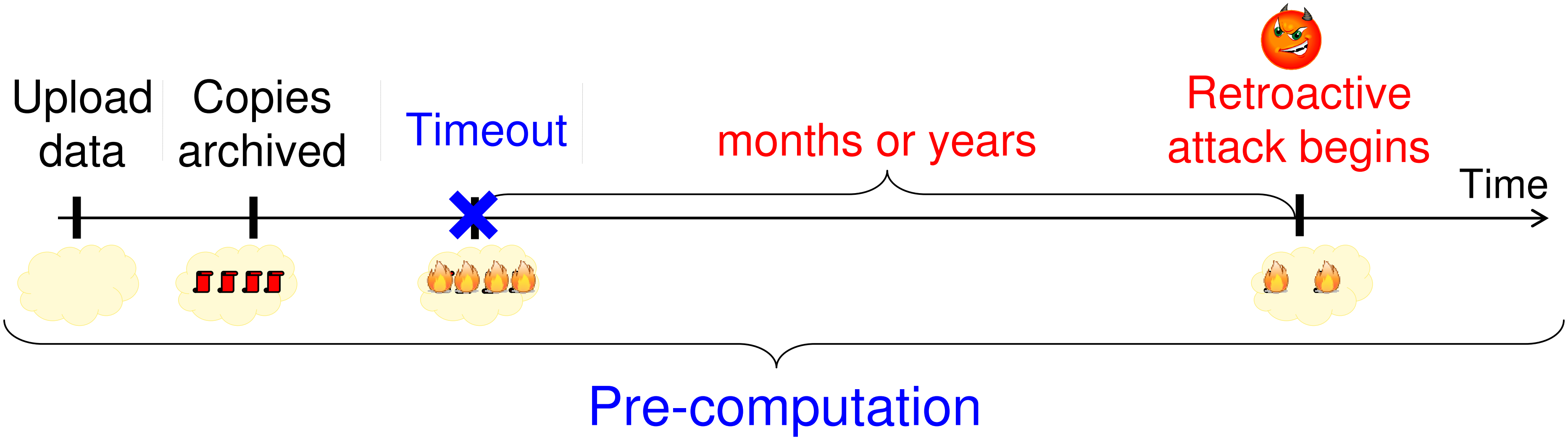
In the paper

Discussed next

- Highest-level results:
  - Secret sharing parameters (N and M) affect availability, timeout, performance, and security
  - Tradeoffs are necessary

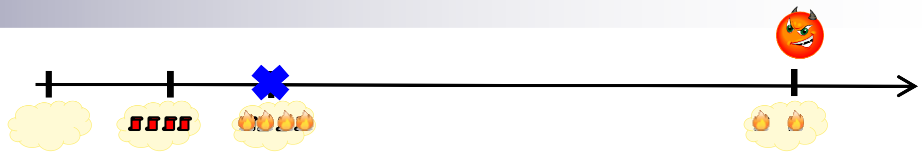
Threat Model

- Goal: protect against retroactive attacks on old copies
  - Attackers don't know their target until after timeout
  - Attackers may do non-targeted "pre-computations" at any time



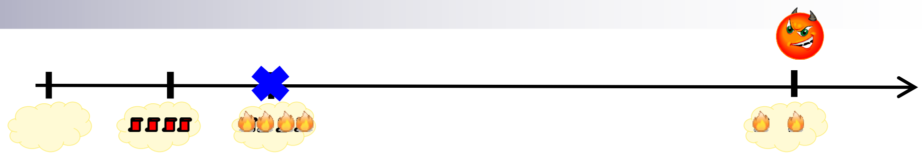
- Communicating parties trust each other
  - E.g., Ann trusts Carla not to keep a plain-text copy

Attack Analysis

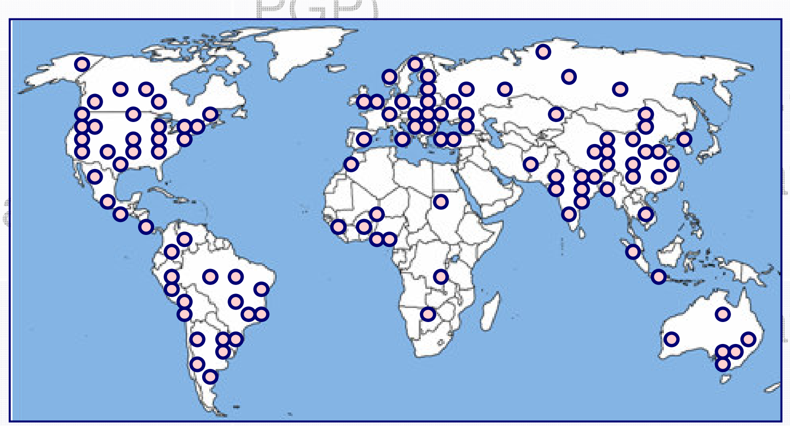


Retroactive Attack	Defense
Obtain data by legal means (e.g., subpoenas)	P2P properties: constant evolution, geographic distribution, decentralization
Gmail decapsulates all VDO emails	Compose with traditional encryption (e.g., PGP)
ISP sniffs traffic	Anonymity systems (e.g., Tor)
DHT eclipse, routing attack	Defenses in DHT literature (e.g., constraints on routing table)
DHT Sybil attack	Defenses in DHT literature; Vuze offers some basic protection
Intercept DHT “get” requests & save results	Vanish obfuscates key share lookups
Capture key pieces from the DHT (pre-computation)	P2P property: huge scale
More (see paper)	

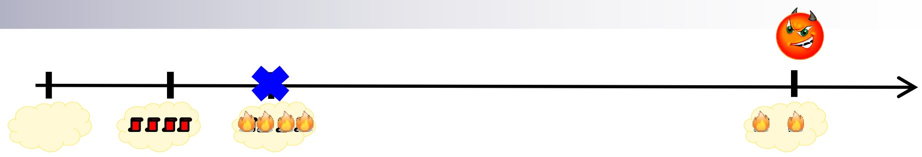
Attack Analysis



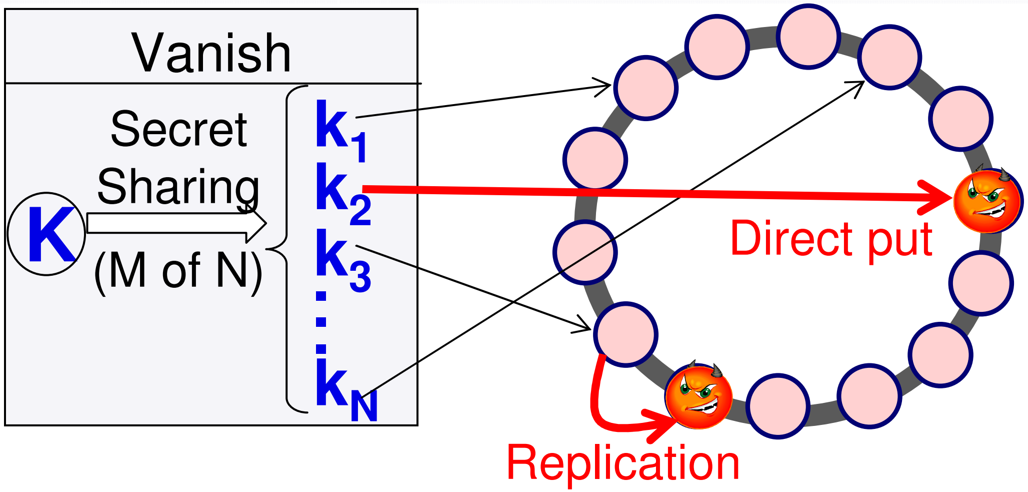
Retroactive Attack	Defense
Obtain data by legal means (e.g., subpoenas)	P2P properties: constant evolution, geographic distribution, decentralization
Gmail decapsulates all VDO emails	Compose with traditional encryption (e.g., PGP)
ISP sniffs traffic	g., Tor)
DHT eclipse, routing a	ture (e.g., constraints
DHT Sybil attack	ture; Vuze offers
Intercept DHT “get” requests & save results	Vanish obfuscates key share lookups
Capture key pieces from the DHT (pre-computation)	P2P property: huge scale
More (see paper)	



Retroactive Attacks



Attack	Defense
Capture any key pieces from the DHT (pre-computation)	P2P property: huge scale



- Given the huge DHT scale, how many nodes does the attacker need to be effective?
- Current estimate:
  - □ Attacker must join with ~8% of DHT size, for 25% capture
  - □ There may be other attacks (and defenses)

Vanish Applications

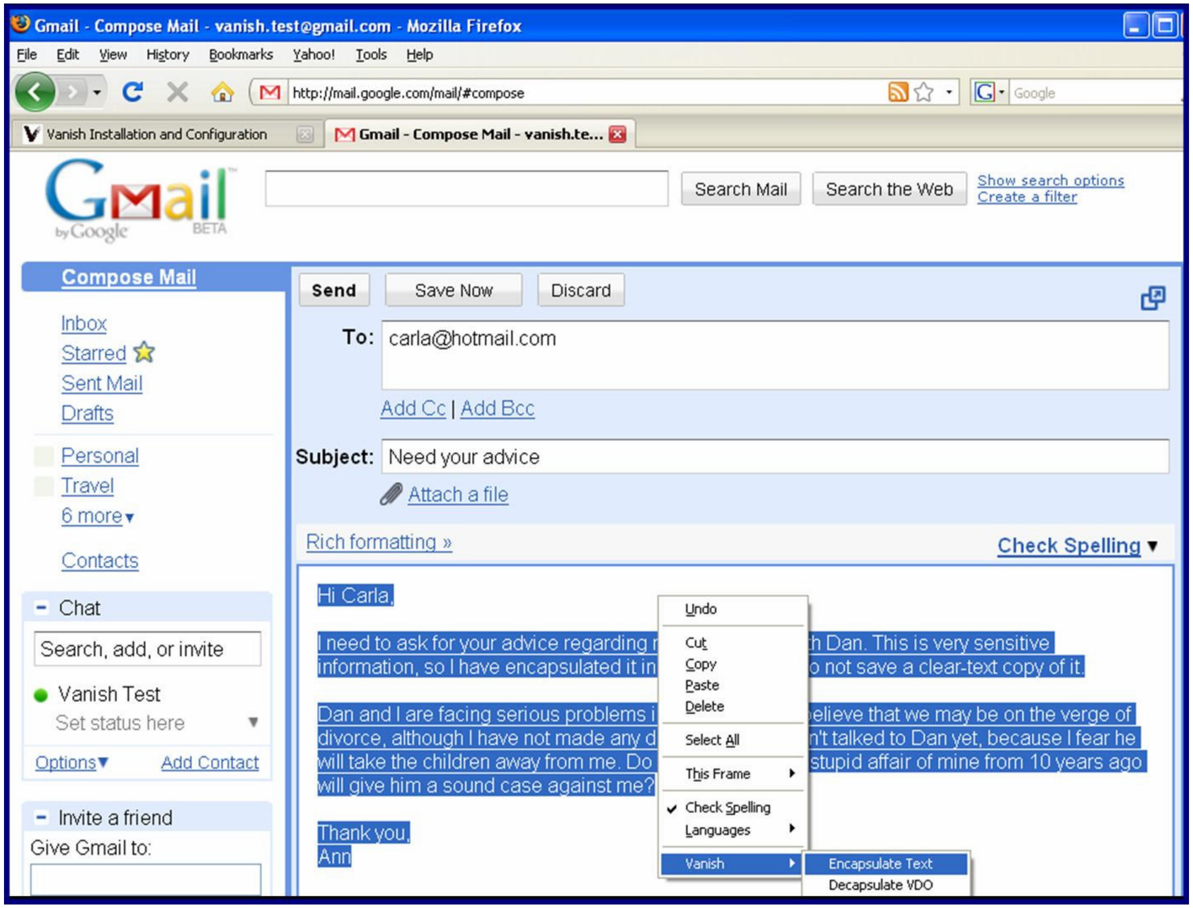
- Self-destructing data & Vanish support many applications

Example applications:

- Firefox plugin
  - Included in our release of Vanish
- Thunderbird plugin
  - Developed by the community two weeks after release ☺
- Self-destructing files
- Self-destructing trash-bin
- …

Firefox Plugin For Vanishing Web Data

- Encapsulate text in any text area in self-destructing VDOs



To: carla@hotmail.com

Subject: Need your advice

Hi Carla,

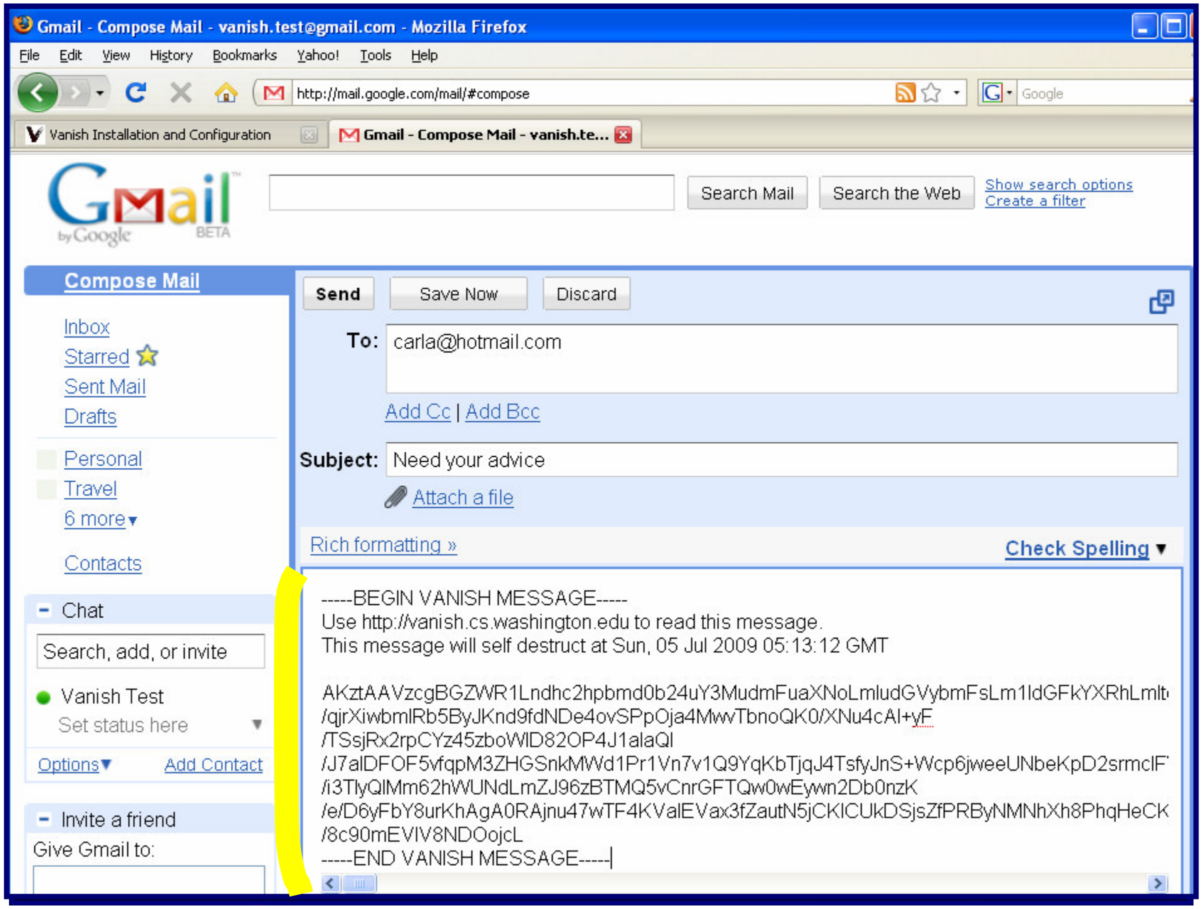
I need to ask for your advice regarding r h Dan. This is very sensitive
information, so I have encapsulated it in o not save a clear-text copy of it.

Dan and I are facing serious problems i elieve that we may be on the verge of
divorce, although I have not made any d n't talked to Dan yet, because I fear he
will take the children away from me. Do stupid affair of mine from 10 years ago
will give him a sound case against me?

Thank you,
Ann

Firefox Plugin For Vanishing Web Data

- Encapsulate text in any text area in self-destructing VDOs



To: carla@hotmail.com

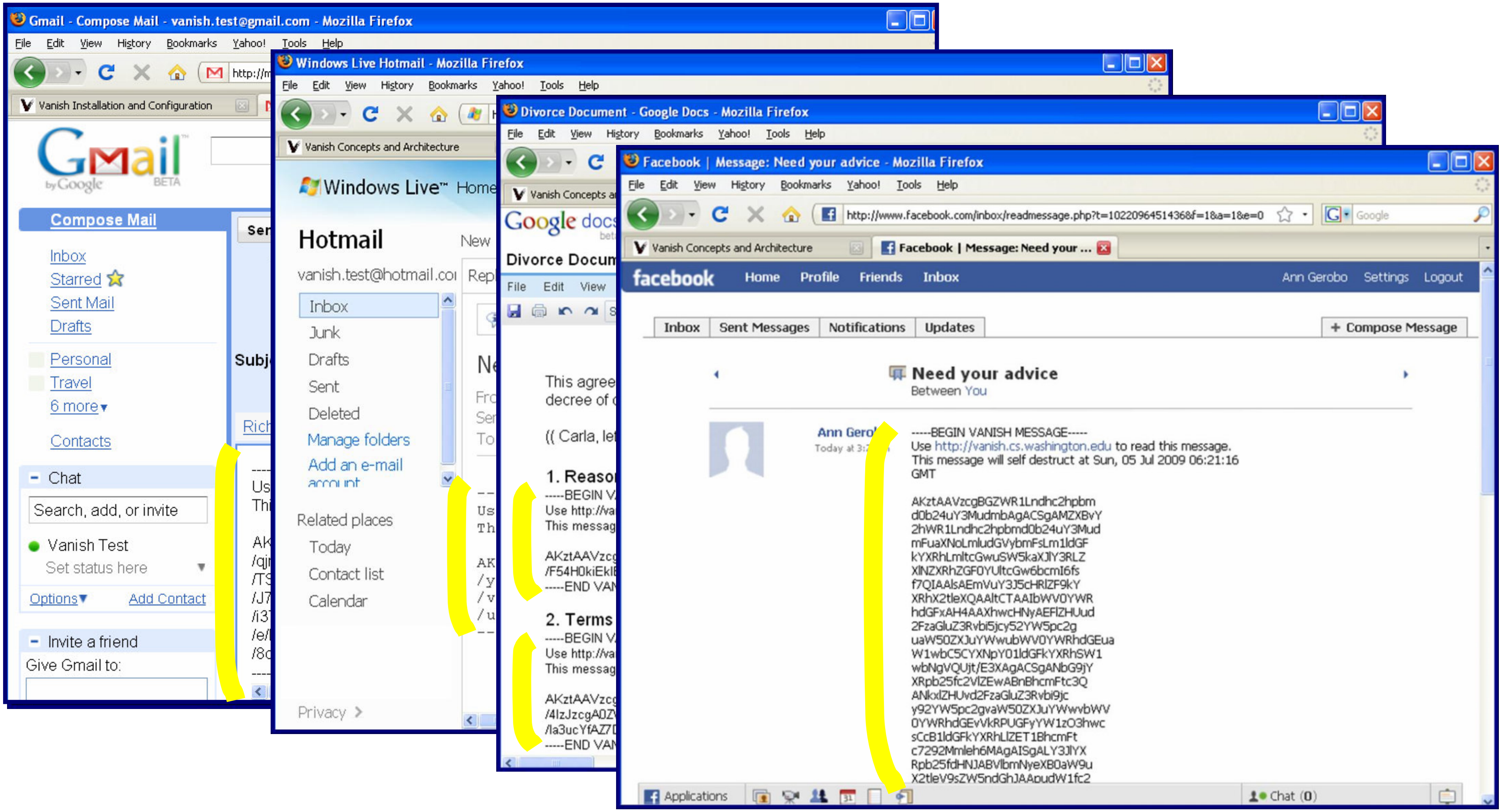
Subject: Need your advice

```
-----BEGIN VANISH MESSAGE-----
Use http://vanish.cs.washington.edu to read this message.
This message will self destruct at Sun, 05 Jul 2009 05:13:12 GMT

AKztAAVzcgBGZWR1Lndhc2hpbmd0b24uY3MudmFuaXNoLmludGVybmFsLm1ldGFkYXRhLmlt
/qjrXiwbmlRb5ByJKnd9fdNDe4ovSPpOja4MwvTbnoQK0/XNu4cAI+yF
/TSsjRx2rpCYz45zboWID82OP4J1alaQl
/J7alDFOF5vfqpM3ZHGSnkMWd1Pr1Vn7v1Q9YqKbTjqJ4TsfyJnS+Wcp6jweeUNbeKpD2srmclF
/i3TlyQlMm62hWUNdLmZJ96zBTMQ5vCnrGFTQw0wEywn2Db0nzK
/e/D6yFbY8urKhAgA0RAjnu47wTF4KValEVax3fZautN5jCKICUkDSjsZfPRByNMNhXh8PhqHeCK
/8c90mEVIV8NDOojcL
-----END VANISH MESSAGE-----
```

Firefox Plugin For Vanishing Web Data

- Encapsulate text in any text area in self-destructing VDOs



Need your advice
Between You

Ann Gero
Today at 3:

-----BEGIN VANISH MESSAGE-----
Use http://vanish.cs.washington.edu to read this message.
This message will self destruct at Sun, 05 Jul 2009 06:21:16 GMT

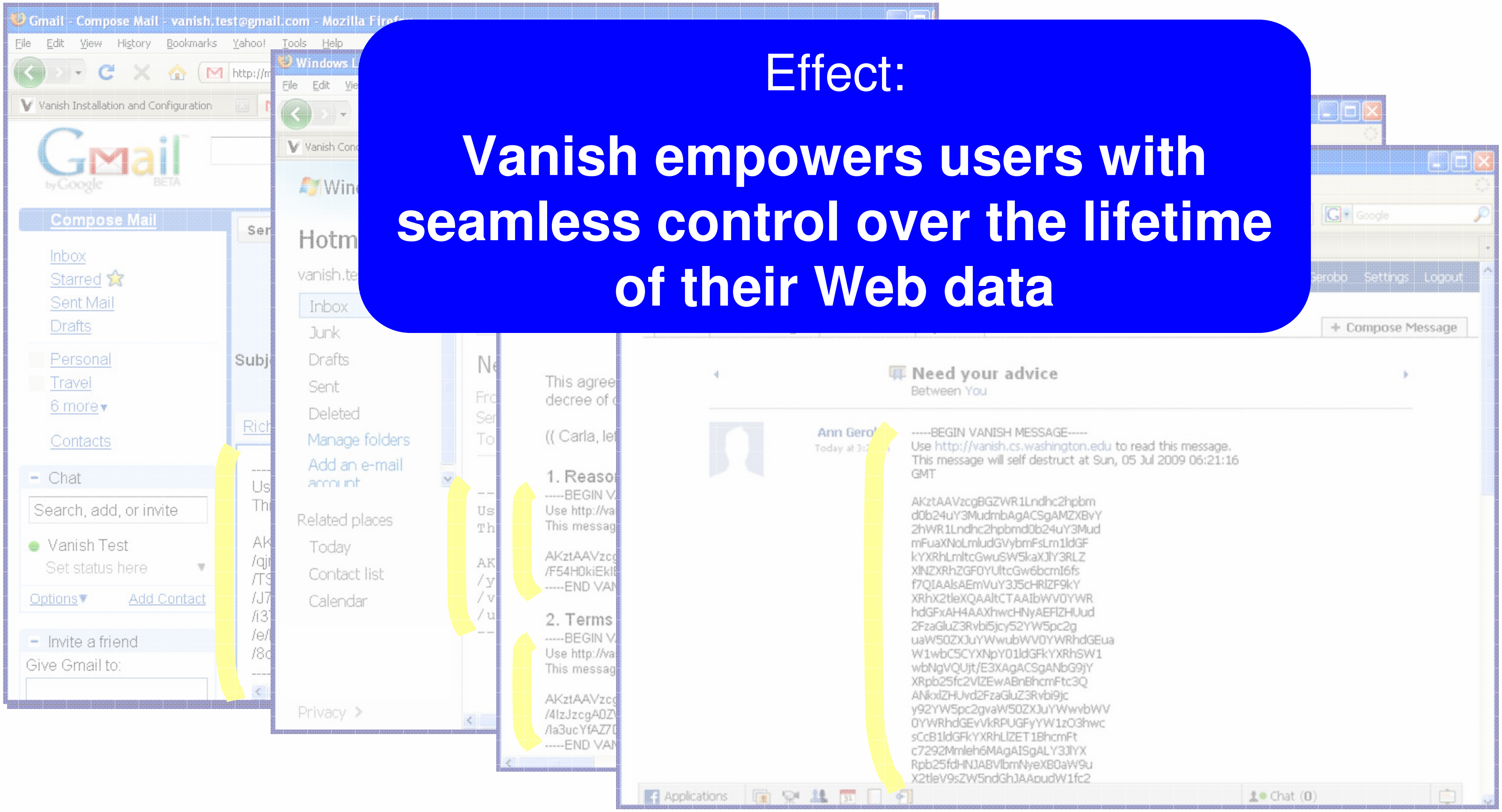
```
AKztAAVzcgBGZWR1Lndhc2hpbm
d0b24uY3MudmbAgACSgAMZXBvY
2hWR1Lndhc2hpbmd0b24uY3Mud
mFuaXNoLmludGVybmFsLm1ldGF
kYXRhLmltcGwuSW5kaXJlY3RLZ
XlNZXRhZGF0YUltcGw6bcmI6fs
f7QIAAlsAEmVuY3J5cHRlZF9kY
XRhX2tleXQAAltCTAAIbWV0YWR
hdGFxAH4AAXhwcHNyAEFlZHUud
2FzaGluZ3Rvbi5jcy52YW5pc2g
uaW50ZXJuYWwubWV0YWRhdGEua
W1wbC5CYXNpY01ldGFkYXRhSW1
wbNgVQUjt/E3XAgACSgANbG9jY
XRpb25fc2VlZEwABnBhcmFtc3Q
ANkxlZHUvd2FzaGluZ3Rvbi9jc
y92YW5pc2gvaW50ZXJuYWwvbWV
0YWRhdGEvVkRPUGFyYW1zO3hwc
sCcB1ldGFkYXRhLlZET1BhcmFt
c7292Mmleh6MAgAISgALY3JlYX
Rpb25fdHNJABVlbmNyeXB0aW9u
X2tleV9sZW5ndGhJAApudW1fc2
```

-----BEGIN V
Use http://va
This messag
AKztAAVzcg
/F54H0kiEkl
-----END VAN

-----BEGIN V
Use http://va
This messag
AKztAAVzcg
/4lzJzcgA0Z
/la3ucYfAZ7
-----END VAN

Firefox Plugin For Vanishing Web Data

- Encapsulate text in any text area in self-destructing VDOs



Effect:

Vanish empowers users with seamless control over the lifetime of their Web data

Need your advice

Between You

Ann Gero

-----BEGIN VANISH MESSAGE-----
Use http://vanish.cs.washington.edu to read this message.
This message will self destruct at Sun, 05 Jul 2009 06:21:16 GMT

AKztAAVzcgBGZWR1Lndhc2hpbm
d0b24uY3MudmbAgACSgAMZXBvY
2hWR1Lndhc2hpbmd0b24uY3Mud
mFuaXNoLmludGVybmFsLm1ldGF
kYXRhLmltcGwuSW5kaXJlY3RLZ
XlNZXRhZGF0YUltcGw6bcmI6fs
f7QIAAlsAEmVuY3J5cHRlZF9kY
XRhX2tleXQAAltCTAAIbWV0YWR
hdGFxAH4AAXhwcHNyAEFlZHUud
2FzaGluZ3Rvbi5jcy52YW5pc2g
uaW50ZXJuYWwubWV0YWRhdGEua
W1wbC5CYXNpY01ldGFkYXRhSW1
wbNgVQUjt/E3XAgACSgANbG9jY
XRpb25fc2VlZEwABnBhcmFtc3Q
ANkxlZHUvd2FzaGluZ3Rvbi9jc
y92YW5pc2gvaW50ZXJuYWwvbWV
0YWRhdGEvVkRPUGFyYW1zO3hwc
sCcB1ldGFkYXRhLlZET1BhcmFt
c7292Mmleh6MAgAISgALY3JlYX
Rpb25fdHNJABVlbmNyeXB0aW9u
X2tleV9sZW5ndGhJAApudW1fc2

Use http://va
This messag
-----END VAN

http://vanish.cs.washington.edu/

Conclusions

- Two formidable challenges to privacy:
  - □ Data lives forever
  - □ Disclosures of data and keys have become commonplace
- Self-destructing data empowers users with lifetime control
- Vanish:
  - □ Combines global-scale DHTs with secret sharing to provide self-destructing data
  - □ Firefox plugin allows users to set timeouts on text data anywhere on the web
- Vanish ≠ Vuze-based Vanish
  - □ Customized DHTs, hybrid approach, other P2P systems
  - □ Further extensions for security in the paper