Presented By: Cole Sherer

Vanish: Increasing Data Privacy with Self-Destructing Data

Problem

- Personal Data Privacy
- Third Party Archiving
- Carelessness
- Theft
- Legal Action

Where is this Problem?

- Corporate / Private Emails
- FaceBook Messages
- Google Docs
- Flickr
- Text Messages

Solution

- Delete all Copies after Certain Time
- No User Action to Delete
- Impossible to Decrypt after Expiration

Contributions

- Identify Requirements for Self-Destructing Data
- Implement System using DHTs
- Demonstrate Correctness of System
- Evaluate the Implementation

Possible Approaches

- Manual Deletion / Cron Job
  - What if Machine is Stolen?
- Traditional Encryption
  - Attacker Gets Key
- Deny Existence of / Hide Data
  - Computationally Implausible
- Trusted 3rd Party Escrow
  - Users are Wary to Trust Service

Goals

- Complete Destruction After Timeout
- Remain Accessible Until Timeout
- Leverage Existing Infrastructures
- No Secure Hardware
- No New Privacy Risks
- No Passwords

Assumptions

- Time-Limited Value
  - Data Not Important After Timeout
- Known Timeout
- Internet Connectivity
  - Required for Encrypt & Decrypt
- Dispensable Under Attack
  - Expire Early
- Trusted Data Owners
- Attacks are Retroactive

Vanish Architecture

- Uses DHTs
  - Allow Huge Name Space (2^{160})
  - Availability
  - Decrentralization
  - Churn

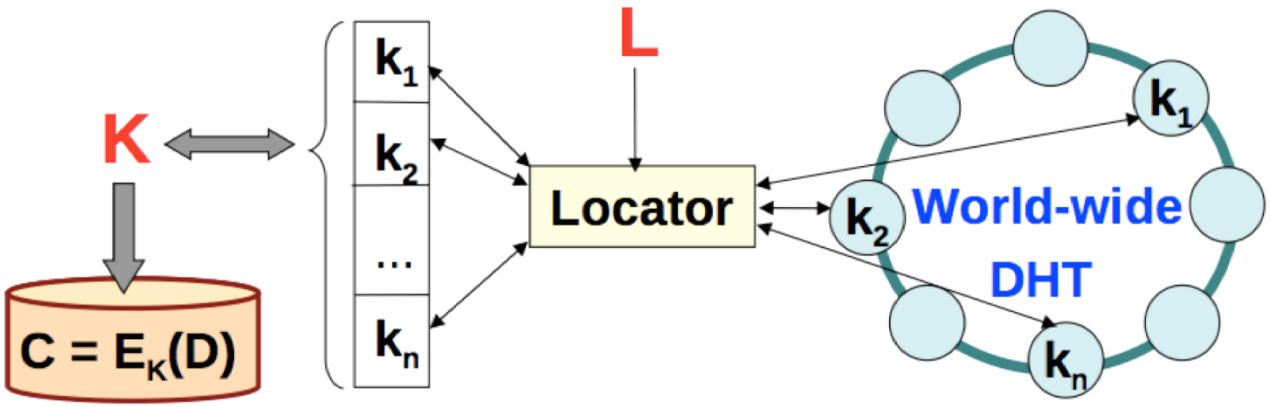
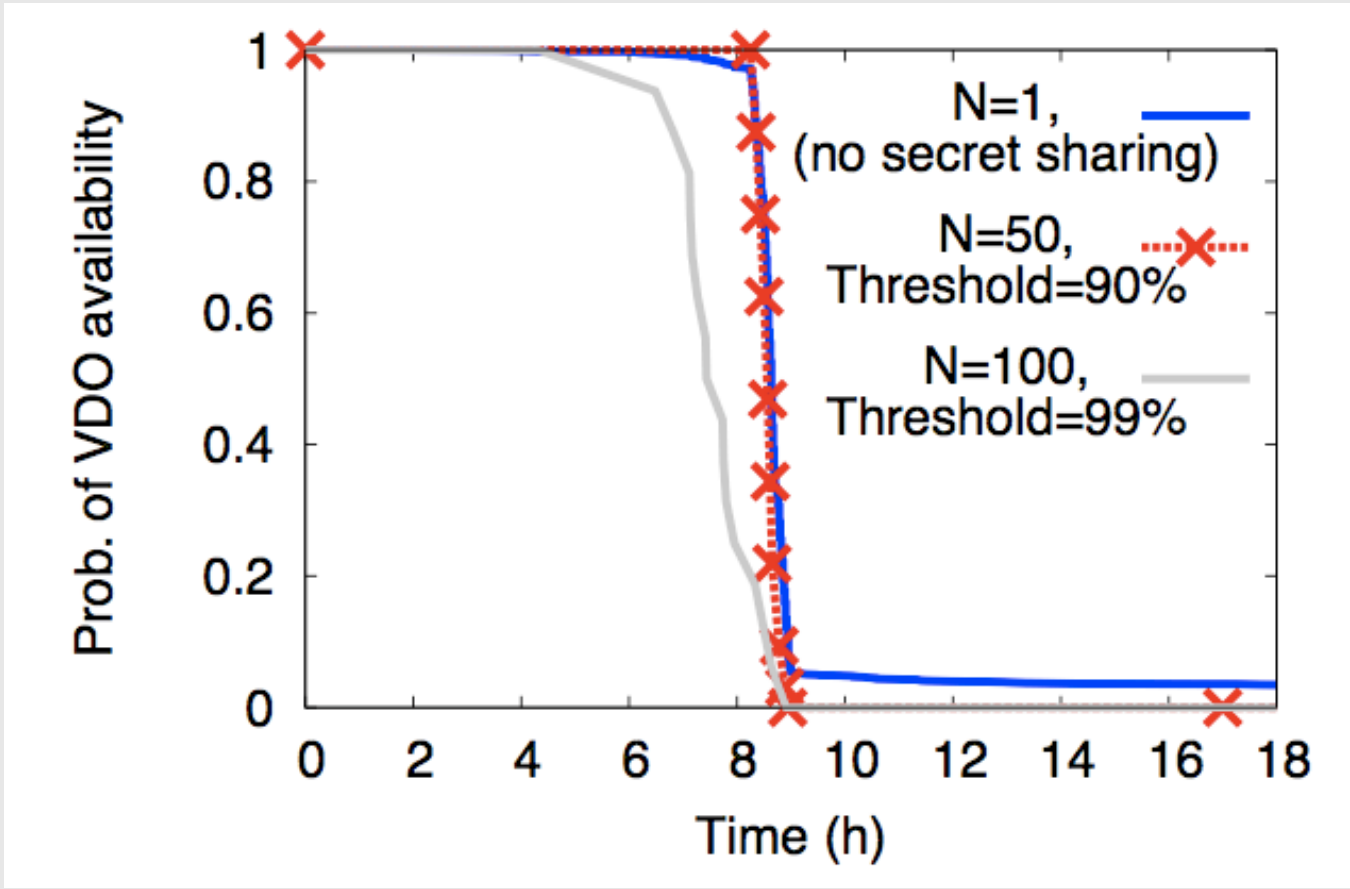


Figure 3: **The Vanish system architecture.**

Vuze (Azureus)



FireVanish

(a) Vanishing Facebook messages.

- GPG Based
- Gmail
- Web Input

Performance

N	Time (seconds)		
	Encapsulate VDO		Decapsulate VDO
	Without prepush	With prepush	
10	10.5	0.082	0.9
20	16.9	0.082	2.0
50	32.8	0.082	4.7
100	64.5	0.082	9.2
150	94.7	0.082	14.0
200	124.3	0.082	19.0

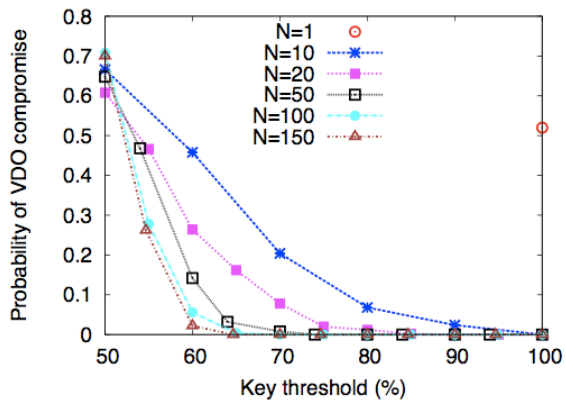
(b) VDO operation execution times.

- Most Execution Time from DHT
- Larger Data == More Encryption & Decryption Time

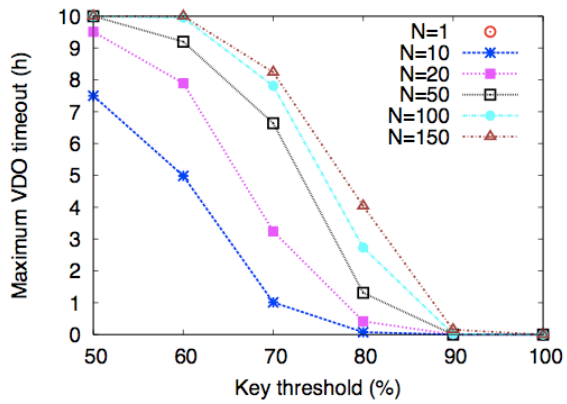
Attack Strategies

- Decapsulate VDO Prior to Expiration
  - Encrypt VDO Using PGP or GPG
- Sniff User's Internet Connection
  - Compose with TOR
- Integrate into the DHT
  - Too Costly ($860K / year)

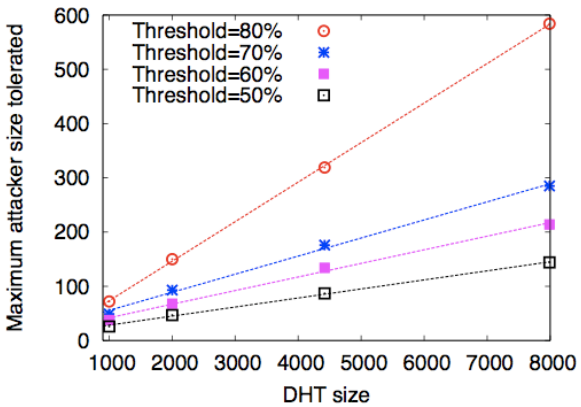
Security Analysis



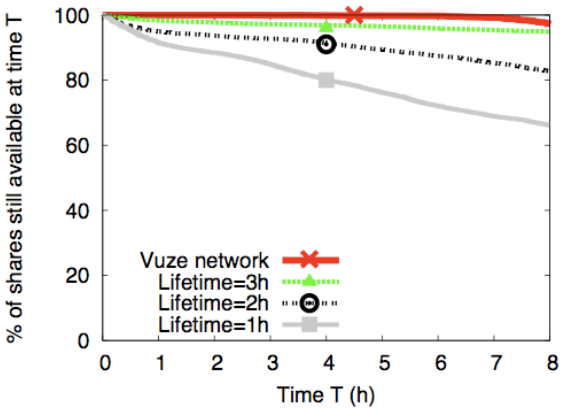
(a) Parameters and security.



(b) Parameters and availability.



(c) Tolerated attacker sizes.



(d) Churn effect on availability.

Conclusion

- Implemented Prototype
- Uses DHTs and Key Sharing
- Files Irreversibly Self-Destruct
- Relatively Efficient
- Safe From Attack