Blindfold: A System to "See No Evil" in Content Discovery

Ryan S. Peterson **Bernard Wong** Emin Gün Sirer

Cornell University

Content Discovery Landscape

- **Mininova** forced to remove all but a small white-list of torrents
- **Pirate Bay** found guilty of assistance to copyright infringement
- **YouTube** sued by Viacom for a billion dollars

Blindfold Goals

- Implement a publicly searchable content discovery service
  - Support non-authenticated clients
- Protect the server from the content
- Empower storage operators to operate as utility providers

Related Work

- Encrypted database search
  - Prevents the database from knowing the contents of the data while providing a search primitive over the encrypted data
  - Requires clients to have out-of-band access to the shared secret
- Not applicable to public content discovery services

Blindfold Approach

- Decouple the content discovery service into two components
  - Index server
  - Content server
- Map keywords to content in a way that a human can navigate, but is difficult to automate

Formal Definition

- Implement a mapping function f(x) → y
- Deconstruct the function f(x) into
  - f(x) = f″(captcha(f′(x)))
  - f′(x) → z; captcha(z) → z′; f″(z′) → y
- Store f′(*) and f″(*) on two key-value servers
  - Optionally in different administrative domains

Upload New Content

Keywords:
- Ubuntu
- Linux
- 9.10

Ubuntu :

Key	Value
Fedora	101010…
Slackware	001010…
Debian	011100…

Upload New Content

Keywords:
- Ubuntu
- Linux
- 9.10

finding : finding

Index server

Key	Value
h^{α}(Fedora)	 , h^{α}(…)
h^{α}(Slackware)	 , h^{α}(…)
h^{α}(Debian)	 , h^{α}(…)

Content server

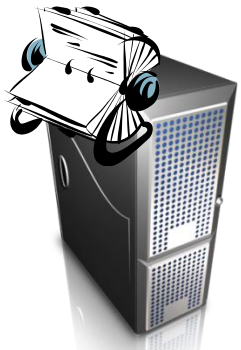
Key	Value
h(kbpsh)	{101010…} $h^{\alpha-1}$(…)
h(3m573)	{001010…} $h^{\alpha-1}$(…)
h(vzpk2z)	{011100…} $h^{\alpha-1}$(…)

Upload New Content

Keywords:
- Ubuntu
- Linux
- 9.10

finding : finding

$h^{\alpha}(Ubuntu) : <$ finding $, h^{\alpha}(\phi) >$



Index server

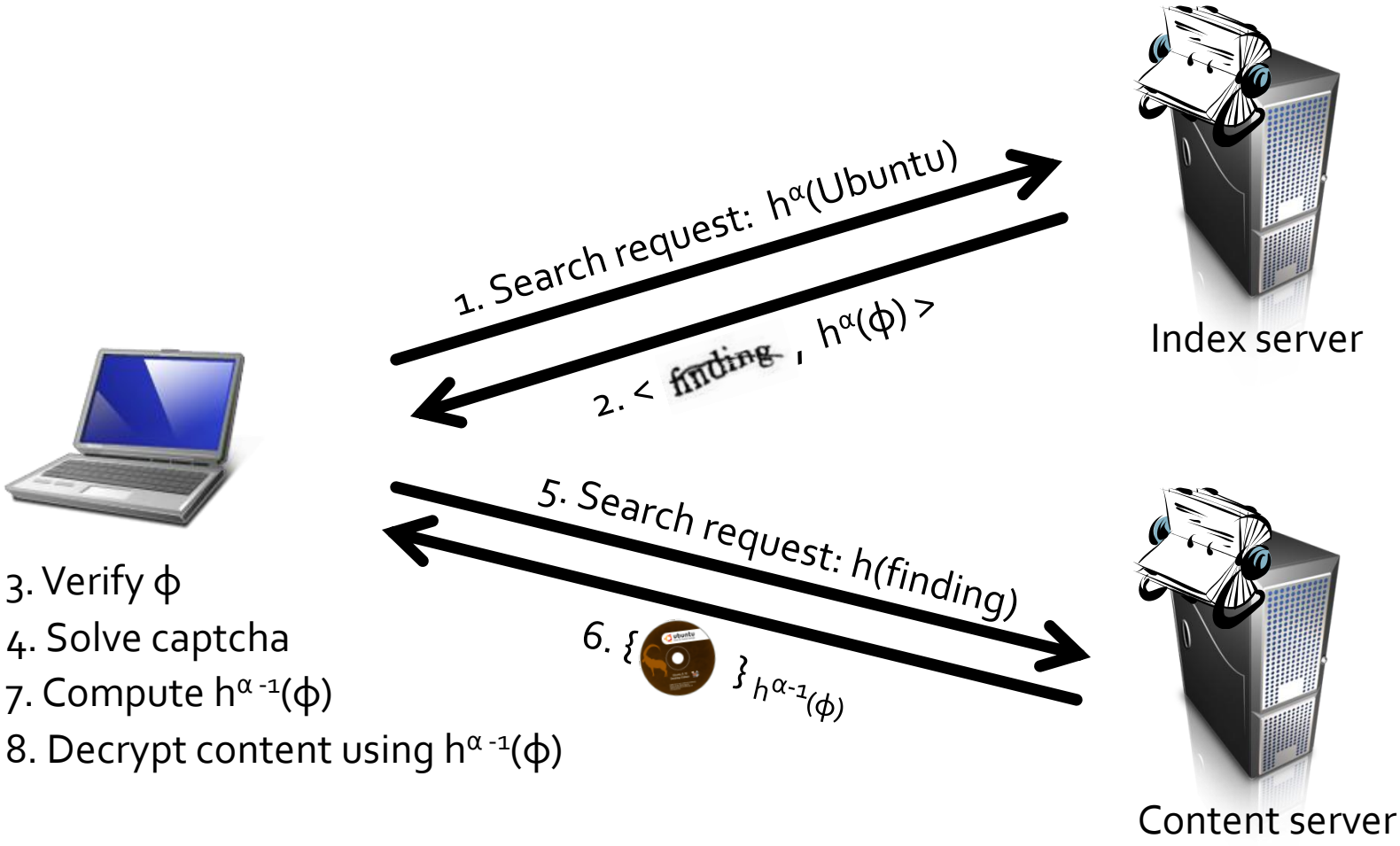
$h(finding) : \{$  $\}_{h^{\alpha-1}(\phi)}$

Content server

$\Phi = hmac_{Ubuntu}($ finding $)$

(used to detect tempering of key-value mapping)

Content Search



Problem: Key Squatting

- A malicious user can squat on a keyword by inserting an unsolvable captcha into the index server
- Solution 1
  - Index server verifies the source of captchas
- Solution 2
  - Store a separate user-generated captcha for each content object under the same keyword
  - Users must solve a captcha for each matching content

Summary

- Fast, efficient, and non-intrusive technique for accessing data through explicit keyword search without revealing the key or the value to the server
- Enables key-value storage services to be completely blind to the content
- Provides plausible deniability to the storage operators