SBA Research

Object Capabilities and Their Benefits for Web Application Security

IKT-Sicherheitskonferenz

Michael Koppmann – 2022-09-15

Bundesministerium Klimaschutz, Umwelt, Energie, Mobilität, Innovation und Technologie

Bundesministerium Digitalisierung und Wirtschaftsstandort

FFG Forschung wirkt.

wirtschafts agentur wien Ein Fonds der Stadt Wien

European Commission

FWF Der Wissenschaftsfonds.

netidee OPEN INNOVATIONS

Motivation

- Vulnerabilities cost companies multiple billion U.S. dollars a year
- Some vulnerabilities are more common than others
- OWASP publishes "Top Ten" list
  - A01:2021-Broken Access Control
  - A03:2021-Injection
  - A07:2021-Identification and Authentication Failures

Access Control Lists and Ambient Authority

	/etc/passwd	/home/alice/secret.txt	/home/bob/shared.txt
Alice	{read}	{read, write}	{read}
Bob	{read}	{}	{read, write}
Carol	{read}	{}	{}

Ambient Authority

- Designation and authorization are separate
- Everything can call an object
- The identity of the caller is used for authorization

Object Capabilities and POLA

- Communicable,
- unforgeable,
- token of authority

A capability is a reference to an object, along with an associated set of access rights.

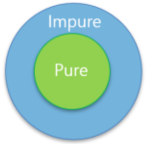
POLA = *Principle of Least Authority*

Research Questions

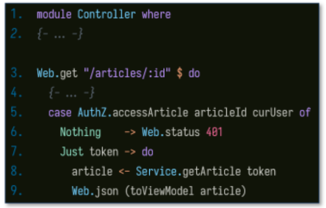
- Can vulnerabilities in authorization systems be prevented by design?
- Is a capability-based system at least as secure as an ACL one?
- Can the web be used as a platform for exchanging security tokens?
- How compatible is this with the rest of the ACL-based ecosystem?

Example: Purely Functional Programming

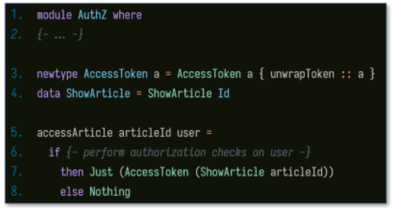
- Pure functions:
  - only rely on input arguments and are deterministic
  - do not perform side-effects like writing to a file or printing to the console, etc.
  - are safe by default as they do not require authority
- Impure functions:
  - perform side-effects but are less composable
  - percentage in code base is kept small to improve auditability
  - authority is granted by passing in arguments



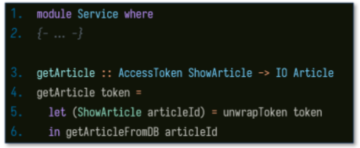
Example: Types as Capabilities



```haskell
module Controller where
{- ... -}

Web.get "/articles/:id" $ do
  {- ... -}
  case AuthZ.accessArticle articleId curUser of
    Nothing    -> Web.status 401
    Just token -> do
      article <- Service.getArticle token
      Web.json (toViewModel article)
```



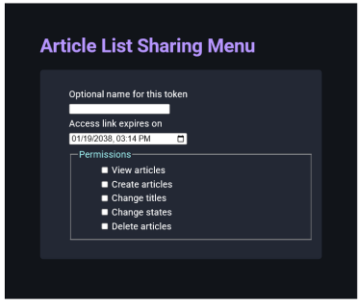
```haskell
module AuthZ where
{- ... -}

newtype AccessToken a = AccessToken a { unwrapToken :: a }
data ShowArticle = ShowArticle Id

accessArticle articleId user =
  if {- perform authorization checks on user -}
    then Just (AccessToken (ShowArticle articleId))
    else Nothing
```



```haskell
module Service where
{- ... -}

getArticle :: AccessToken ShowArticle -> IO Article
getArticle token =
  let (ShowArticle articleId) = unwrapToken token
  in getArticleFromDB articleId
```

Example: URLs as Capabilities

Fine-grained permission levels



Article List Sharing Menu

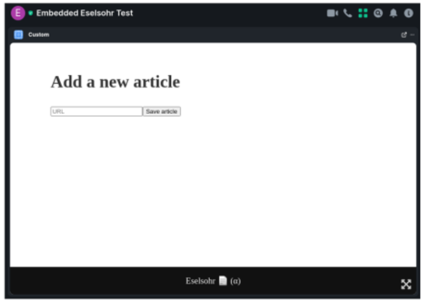
Optional name for this token

Access link expires on

01/19/2038, 03:14 PM

Permissions
- ■ View articles
- ■ Create articles
- ■ Change titles
- ■ Change states
- ■ Delete articles

Embeddability



Add a new article

https://eselsohr.example.org/articles/shared-links?acc=A4LMCKUVYTGCIYVIIWLFWQYGYDKF

Conclusion

- OCAP style programming prevents certain security vulnerabilities by design
- No significant drawbacks compared to ACL could be observed
- Current browsers lack the ability to protect capabilities in URLs:
  - Hyperlinks between web applications can leak capabilities
  - Secure capability transport within the same application is possible
- OCAP style applications do not require specialized frameworks

Master Thesis

https://repositum.tuwien.at/handle/20.500.12708/18849

Michael Koppmann

SBA Research

Floragasse 7, 1040 Vienna, Austria

mkoppmann@sba-research.org

Bundesministerium Klimaschutz, Umwelt, Energie, Mobilität, Innovation und Technologie

Bundesministerium Digitalisierung und Wirtschaftsstandort

FFG Forschung wirkt.

wirtschafts agentur wien Ein Fonds der Stadt Wien

European Commission

FWF Der Wissenschaftsfonds.

netidee OPEN INNOVATIONS