Design and Implementation of Web Forward Proxy with Shibboleth Authentication

KOMURA Takaaki  Kyoto University
SANO Hiroaki  Kyoto University Library
DEMIZU Noritoshi  OCTOPATH corporation
MAKIMURA Ken  OCTOPATH corporation

SAINT 2011 WS (MidArch) @ Munich  2011/07/21
Contents

• Proposal Overview

• Background

• Proposal Details

• Implementation and Evaluations
Proposal Overview

- Shibboleth Authentication introduced into proxy authentication scheme (Proxy-Auth)

Diagram:
- Web Browser
- Shibboleth IdP (Identify Provider)
- Forward Proxy
- Shibboleth SP (Service Provider)
- Web Server

Steps:
1. Try to access via Proxy
2. Authentication by IdP
3. Access via Proxy
4. Proxy relay request
BACKGROUND
Necessity of Proxy and Proxy-Auth

Three reasons

• Gateway from private network to the Internet
• Rapid incident response
• Keep track of access statistics for E-Journal (EJ) sites
  – License fee of EJ will be charged for departments depending on the number of downloading papers

→ Forward Proxy for EJ has been installed in our university since 2006
Authentication to use Forward Proxy

Browser

GET http://example.com/doc

Forward Proxy

407 Proxy Auth Required
Proxy-Authorenticate: Basic realm="XXXXXX"

GET http://example.com/doc
Proxy-Authorize: Basic BASE64ENC==

repeat

Web Server

http://example.com

GET http://example.com/doc

200 OK

GET http://example.com/doc

200 OK
Problems of Existing Proxy-Auth

• BASIC Authentication
  – User ID and password travel in plain text across the network
• Digest Authentication
  – The proxy needs Users’ raw password
    => Security risk is increased

• No method exists to distinguish proxy is real or fake
  – ID and password might be exploited by fake proxy
Purpose and Proposal

Purpose

• More secure Proxy-Auth for users and administrators
• No modification on web browsers  
  – Modifications or plugins are unsuited to practical use

Proposal

• Shibboleth Authentication-capable forward proxy
PROPOSAL DETAILS
Basic Idea

Browser

GET http://example.com/doc

IdP (Identity Provider)

Auth Request (ID & password)

Auth OK

Proxy as a SP (Service Provider)

302 HTTP redirect

Issue session cookie

Set-Cookie: LH741Q...

Check session cookie and relay remaining requests

GET http://example.com/doc

Cookie: LH741Q...

Web Server

http://example.com

GET http://example.com/doc

200 OK

GET http://example.com/doc

200 OK
Session Cookie Restriction

- Browsers send only the cookies which issued by the web server itself
  - The proxy must pretend the web server when the cookies issue (Set-Cookie)
  - The proxy must issue new cookies whenever browser access to new web servers.

$\rightarrow$ Single Sign-On scheme of Shibboleth could avoid bothering for a lot of re-authentications
Ordinary Shibboleth Auth Flow

GET http://example.com/doc/

POST ID and password

POST https://example.com/Shibboleth.sso/SAML2/…

Set-Cookie: LH741Q…

GET http://example.com/doc/
  Cookie: LH741Q…

repeat

SP endpoint
Proposed Auth Flow

Browser

GET http://example.com/doc/

GET https://proxy.net/Shibboleth.sso/Proxy/…

IdP

POST https://proxy.net/Shibboleth.sso/SAML2/…

Forward Proxy

Proxy module

SP module module

GET http://example.com/Shibboleth.sso/Proxy/…

Web Server

GET http://example.com/doc/

Cookie: LH741Q…

repeat

GET http://example.com/doc/

Set-Cookie: LH741Q…

GET http://example.com/Shibboleth.sso/Proxy/…

GET http://example.com/doc/

Cookie: LH741Q…

GET http://example.com/doc/
The Role of New Endpoints

Gather requests to all EJ sites into only one hostname to reduce patterns of SP metadata. “proxy.net” is registered as the SP in this example.

GET http://example.com/doc/
GET https://proxy.net/Shibboleth.sso/Proxy/…
POST https://proxy.net/Shibboleth.sso/SAML2/…
GET http://example.com/Shibboleth.sso/Proxy/
Set-Cookie: LH741Q…
GET http://example.com/doc/
Cookie: LH741Q…

To cope with session cookie restriction
The forward proxy pretends the web server when session cookies is issued (Set-Cookie)
IMPLEMENTATION AND EVALUATIONS
Implementation

• Shibboleth auth capable forward proxy (shibproxy) based on
  – Shibboleth SP 2.4.2
    • 880 lines modification (diff –u style)
    • supports new endpoints
  – Apache 2.2.17
    • Not modified
    • mod_proxy for forward-proxy
    • mod_rewrite for redirection to the new endpoints
Experiments and Results

• Prepare PAC file which directs browser to
  – shibproxy for restricted access EJ sites
  – University’s official anonymous forward proxy for other sites

• Visit several EJ sites by 5 popular browsers
  – IE8, Safari, Firefox, Opera and Chrome

• shibproxy work well
  – User can access EJ sites through shibproxy
  – Authentication is required only once
  – Single Sign-On for ordinary SPs work well
Some Problems and Solutions

• Third party cookie problems
  – Some EJ sites use multiple host name
    e.g. www.example.com and portal.example.com
    (sibling servers under example.com)
  →Send “Set-Cookie” header with
    “domain=.exmaple.com” attribute

• No cookie is sent for some requests
  – favicon.ico
  – OpenSearch
  – pass through the requests whose URL matches regular
    expression (e.g. /favicon\w*[.ico$] )
Future Work

• Support HTTPS
  – Our proposal can not support HTTPS
  – Shibproxy can not intercept cookies in HTTPS session
    → Reverse-Proxy, wildcard certification or modification protocol

• Hybrid Proxy (forward proxy + reverse proxy)
  – HTTP → forward proxy
  – HTTPS → reverse proxy
  – Both can run on one host
  – Both support Shibboleth SSO authentication
Conclusion

• Shibboleth-capable forward proxy
  – We will use the proxy to access to E-Journal sites
  – The proxy pretends the web server when cookies issue

• Some problems and solutions
  – Third party cookie → add “domain” attribute
  – No cookie is sent for some resources
    → pass thorough them specified by REGEXP

• Future work
  – Hybrid forward-reverse proxy for both HTTP and HTTPS
HTTPS Through Forward Proxy

1. Browser sends a CONNECT request to the forward proxy with the destination as `example.com:443`.
2. The forward proxy creates a secure connection with the web server and forwards the SSL encrypted GET request to `http://example.com/doc`.
3. The web server responds with a 200 OK status code.
4. The forward proxy tranfers the response back to the browser.
Phantom URL

Browser

GET http://example.com/doc/

GET https://proxy.net/Shibboleth.sso/Proxy/…

POST https://proxy.net/Shibboleth.sso/SAML2/…

GET http://example.com/Shibboleth.sso/Proxy/…

Set-Cookie: LH741Q…

GET http://example.com/doc/

Web Server

http://example.com

GET http://example.com/doc/

GET https://proxy.net

Redirect to phantom URL

Cookie for the Web Server

Forward Proxy

Proxy module module

https://proxy.net

Redirect to http://example.com/doc/

repeat

Phantom URL