

# Approaches to Simplify Server Authentication

Frederick Hirsch - Nokia  
Hubert A. Le Van Gong – Sun microsystems

W3C Workshop on Transparency and Usability of Web Authentication  
15/16 March 2006 — New York City, USA

# Motivation

- No effective authentication of Service Providers to clients today
- We describe two possible approaches:
  - Shared secret between parties (the service provider & client),
  - Simplified Server Authentication (SSA) – SSO for service providers.

# Shared Secret Approach

Concept - The SP registers itself to the client which creates an “account” and generates a shared secret to be presented by the SP.

Pros - No 3<sup>rd</sup> party is introduced between the SP and the client

## Issues

- Requires standardization of format and sharing of secret
- Heavy lifting done on the client
  - Plugin approach is possible (albeit difficult)
- Potential scalability issue ( $N^2$ )

# Simplified Server Authentication

## Concept

- Focus on relatively simple browser client
- Applying the SSO concept and infrastructure to support Service Provider authentication
- Not discussing web service client
  - This offers additional solutions
    - e.g. Require SP to be authenticated before it can be discovered (e.g. Liberty Alliance discovery model)
    - Support signatures in web service security headers

# SSA Advantages and Issues

## Pros

- Can combine with existing SSO
- Easy to deploy on existing clients
- Re-uses well-established SSO framework with different scenario choices

## Issues

- Some SSO mechanisms introduce more complicated protocol flows
- Some difficult security issues may remain (replay, confidentiality, adversarial SP)

# SSA Approaches

- IDP provides secret
- IDP accessed as portal
- Enhanced Client or Proxy (ECP)

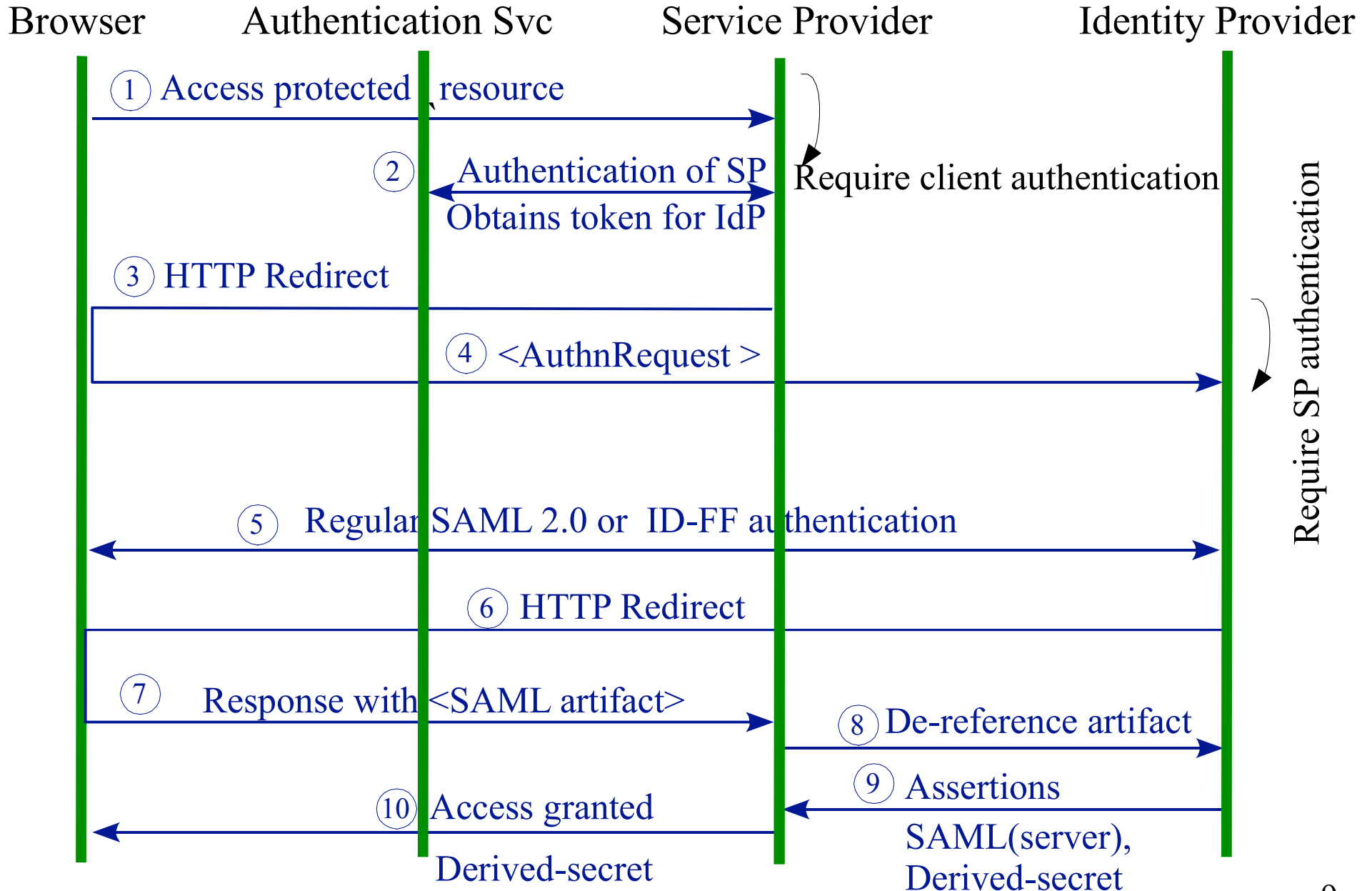
# IDP Secret Approach

- Client expects secret to be provided
  - Client has stored IDP Secret at IDP
  - This is made unique per SP by using SP name
  - Protect against replay, provide confidentiality from SP and others by using hash, including time.
- Client may indicate capability & requirement in request
- In addition, SP knows IDP will not authenticate client unless SP presents SP authentication token, for the cases where SP requires client authentication

# IDP Secret Approach

- SP authenticates (to Authentication Service) and obtains a token to be presented to the IDP.
- Redirect methods of ID-FF or SAML 2.0 can be used to achieve client authentication
- Hash verification is simple, and does not require client signature verification (PKI and general key distribution)
- Issue – requires client to check secret, possible extension or plugin



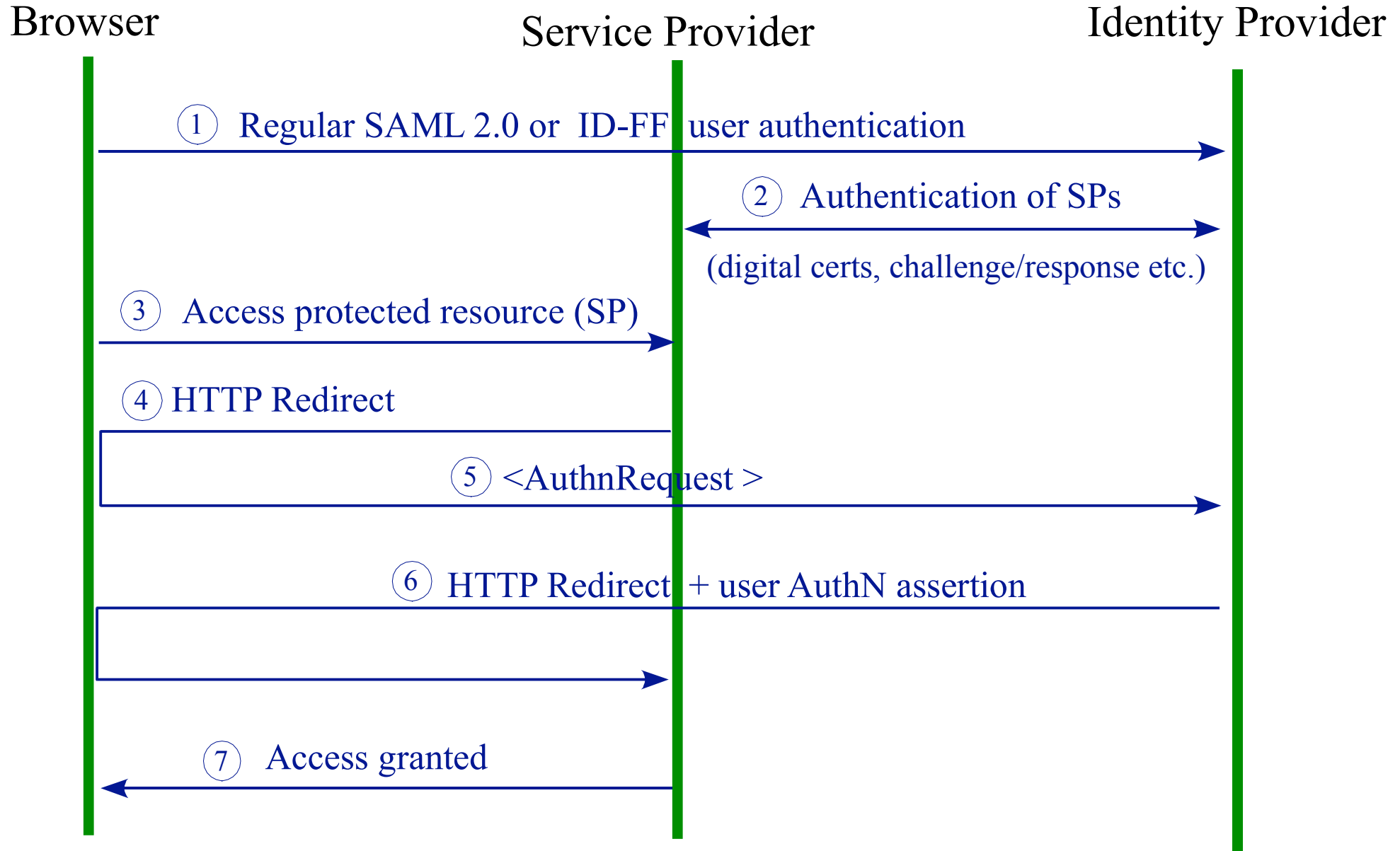


# IDP Accessed As Portal

- Identity provider can in fact also be a service provider and portal
- Client does not need to see shared secret, can rely on trusted IDP in this case
- Requires IDP configuration that SP authentication required

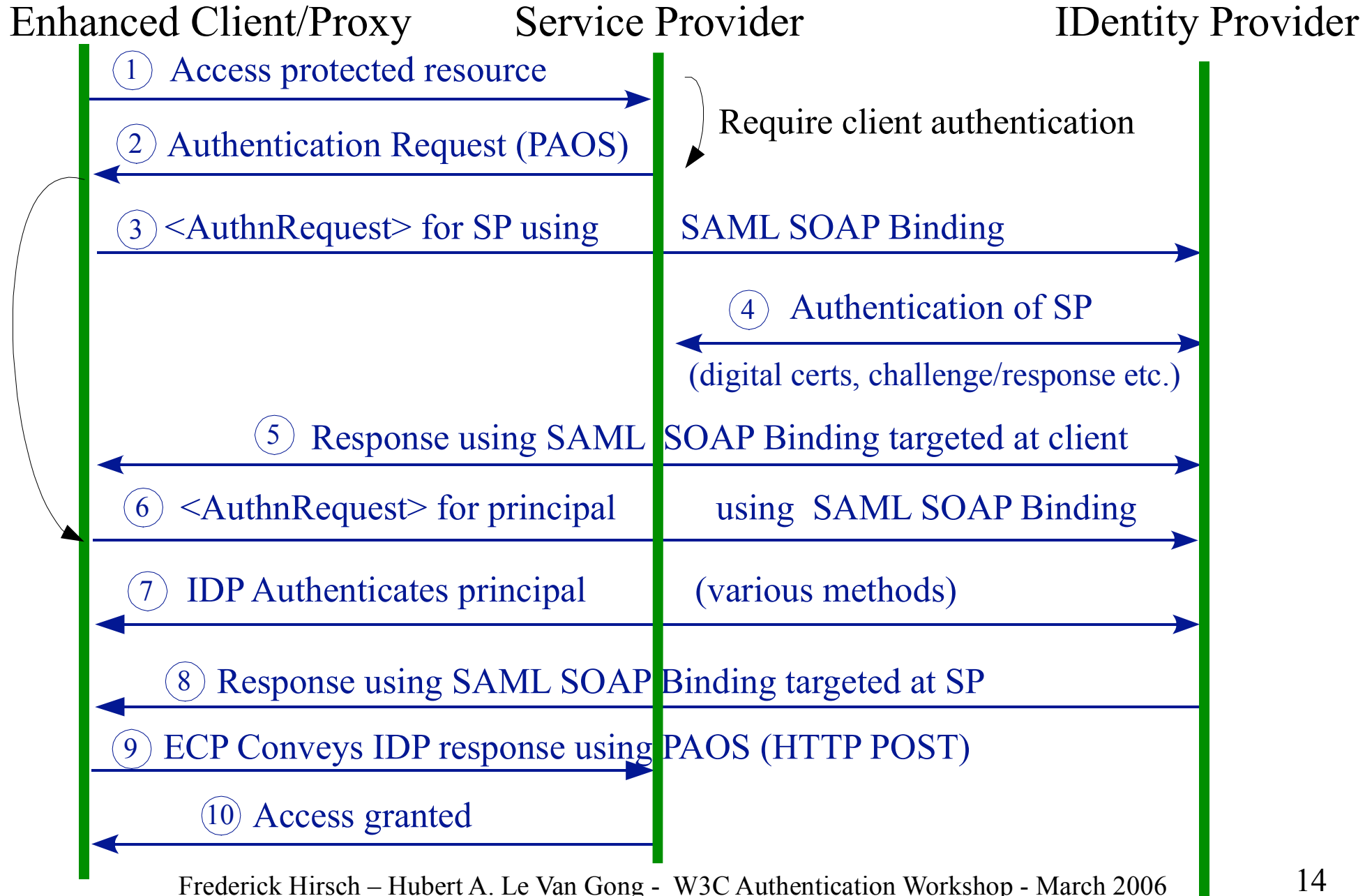
# IDP Portal Operation

- Client authenticates to IDP
- Client then attempts to access another SP in circle of trust using link on portal site
- IDP can require SP authentication before allowing transfer
  - May pre-authenticate portal links
  - May remember recent authentication
- No dependency on SP requiring client authentication



# Enhanced Client or Proxy Approach

- Intelligent client (or proxy) knows how to reach IDP
- Uses SOAP messages conveyed over reverse HTTP binding (PAOS)
- ECP enforces requirement for SP authentication, also actively participates in principal authentication to SP
- Re-uses mechanisms defined in SAML 2.0 standard



# SSA Approaches Summary

| Approach                      | ECP                                                                          | IDP shared secret                                                               | IDP Portal                             |
|-------------------------------|------------------------------------------------------------------------------|---------------------------------------------------------------------------------|----------------------------------------|
| <b>Benefits</b>               | General, active component manages meeting mutual authentication requirements | Scalable shared secret with minimal client changes                              | Trusted intermediary (IDP)             |
| <b>Limitations</b>            | Requires enhanced client or proxy.                                           | Agreement on the representation of the secret and implementation on the client. | Inherent portal limitations            |
| <b>Additional Component?</b>  | Yes (Enhanced client or Proxy)                                               | Liberty Authentication Service technology – ID-FF                               | Liberty ID-FF technology or equivalent |
| <b>Specification Involved</b> | SAML 2.0 ECP or Liberty Alliance LECP                                        | ID-FF & ID-WSF (partial for AS)                                                 | ID-FF                                  |
| <b>Changes to Client?</b>     | No                                                                           | Possibly                                                                        | No                                     |

# References

- Approaches to Simplify Server Authentication
  - <http://www.w3.org/2005/Security/usability-ws/papers/07-nokia-and-sun/>
- SAML 2.0 specifications
  - <http://www.oasis-open.org/apps/org/workgroup/security/#samlv20>
- Liberty specifications
  - <https://www.projectliberty.org/resources/specifications.php>