



## ABSTRACT

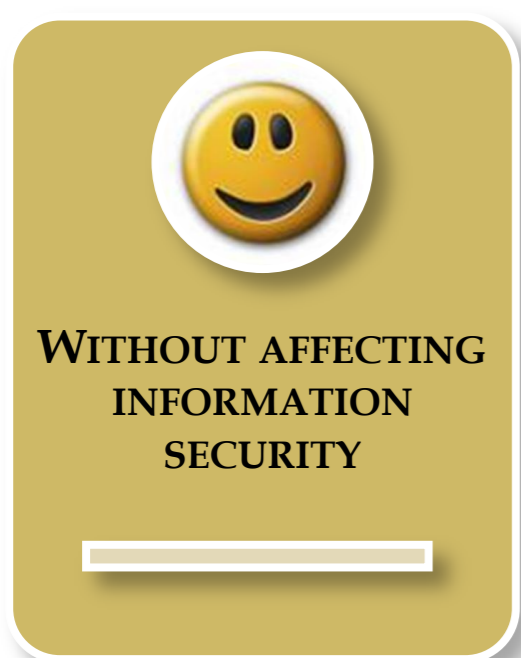
Single Sign On(SSO) architectures for web based applications are gaining widespread adoption in both academic and research institutions but SSO-enabled webmail servers are not, due to potential security issues. The proposed original solution is not affecting information security and is currently being used at the University of Padua.

## Introduction

### Objective

Let user John Smith use the same credentials to gain access to:

- ❑ Webmail through Single Sign On
- ❑ Mail using a POPS client



## Solution

Pass the webmail server as a *shib-attribute* the user's password hash as kept stored in the LDAP authentication backend used by the IdP and the mail server.

In our example the *shib-attribute*:

- ❑ Is not: *cat*
- ❑ But: *H(cat)*  
(es: MD5(*cat*), SHA(*cat*), ...)



The SSO password is not passed to the Webmail server!!



### Secondary LDAP implementation

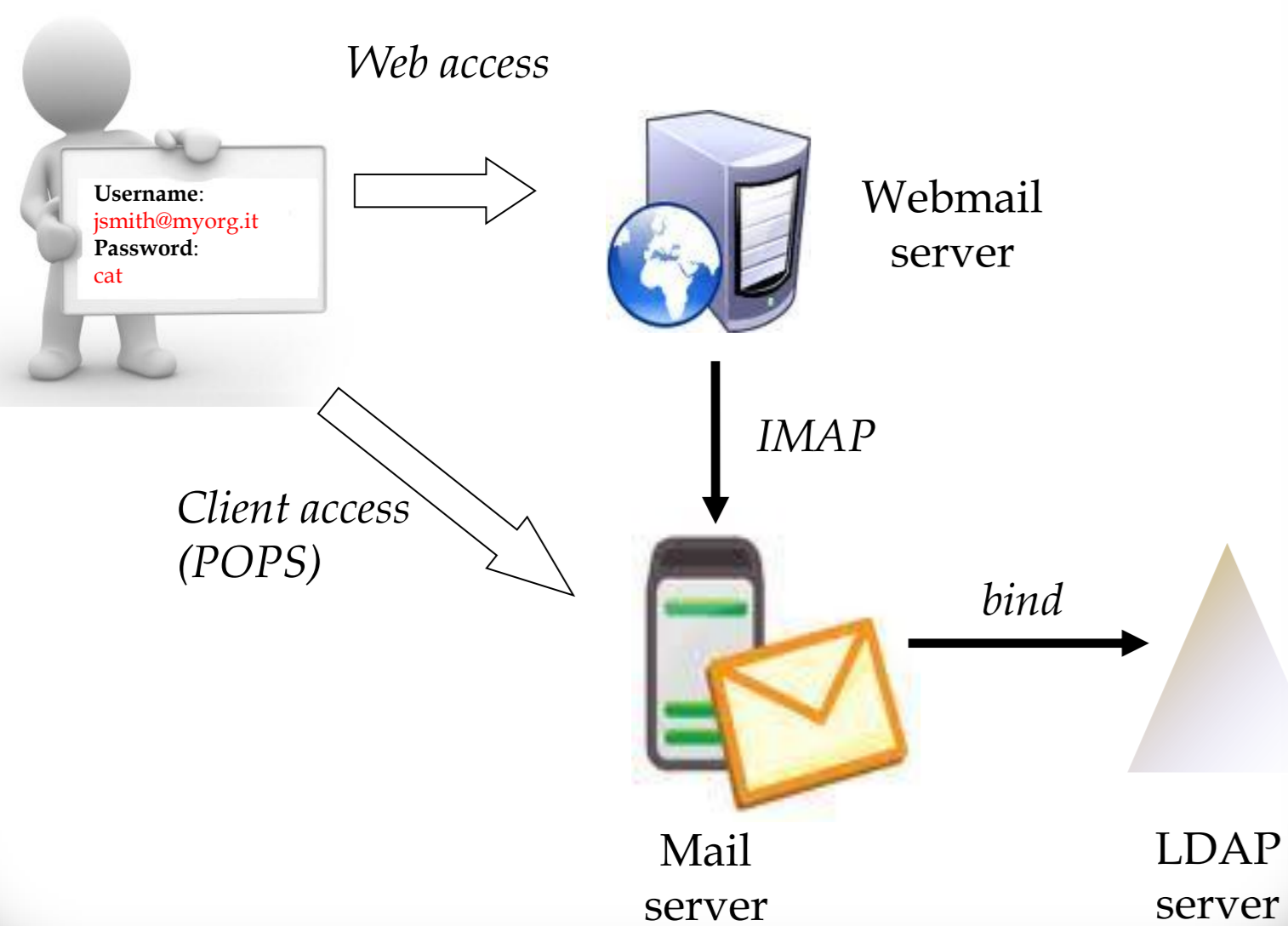
- ❑ Avoid using two *userpassword* attributes - *H(cat)* and *H(H(cat))* - in the same LDAP server!

*H(cat)* would become a second valid SSO password, webmail-cracker's thankfulness guaranteed!

- ❑ A second LDAP server is needed but password changes synchronization has to be managed. Options:

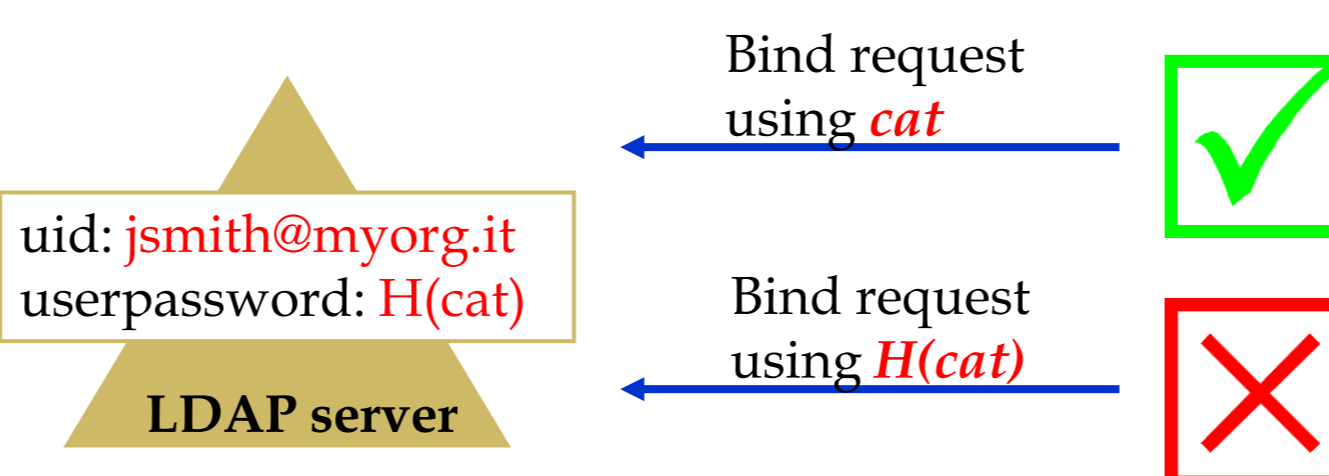
- A real LDAP server (very frequent provisioning needed!)
- Maybe writing an Openldap syncing overlay (coding needed!)
- A Virtual Directory Server (our choice: Penrose).

### Our infrastructure



### One more hurdle

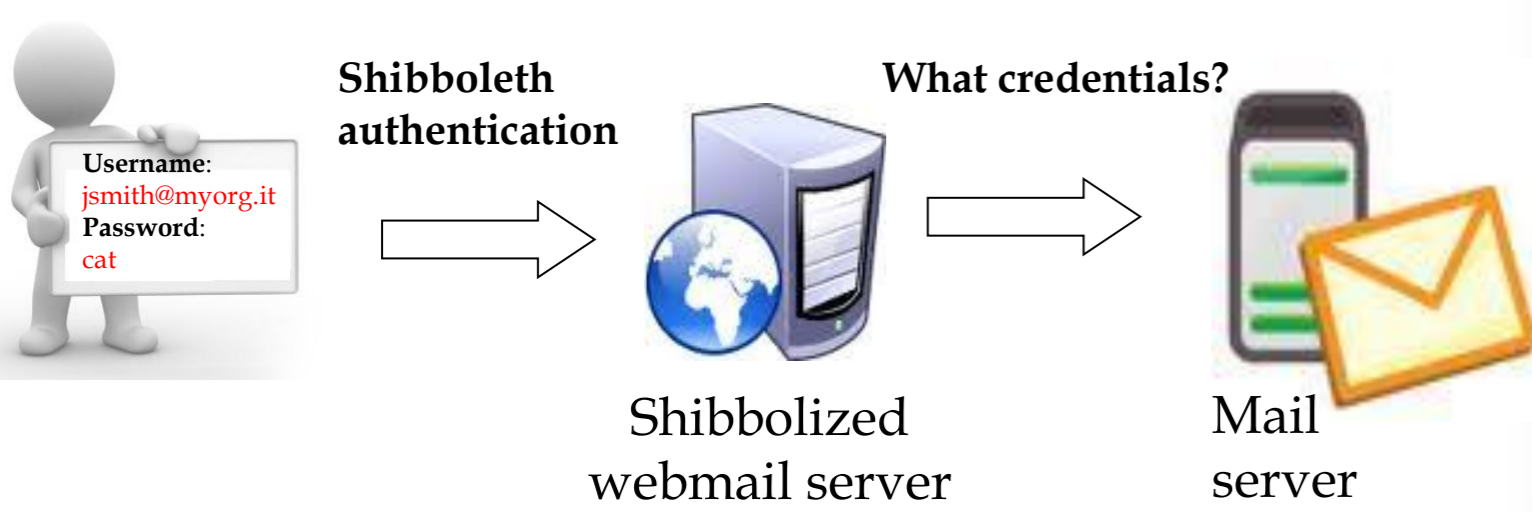
- ❑ The LDAP *userpassword* attribute value is *H(cat)*: bind requests can only be performed using *cat*:



- ❑ How can the mail server perform a bind request knowing only *H(cat)*?

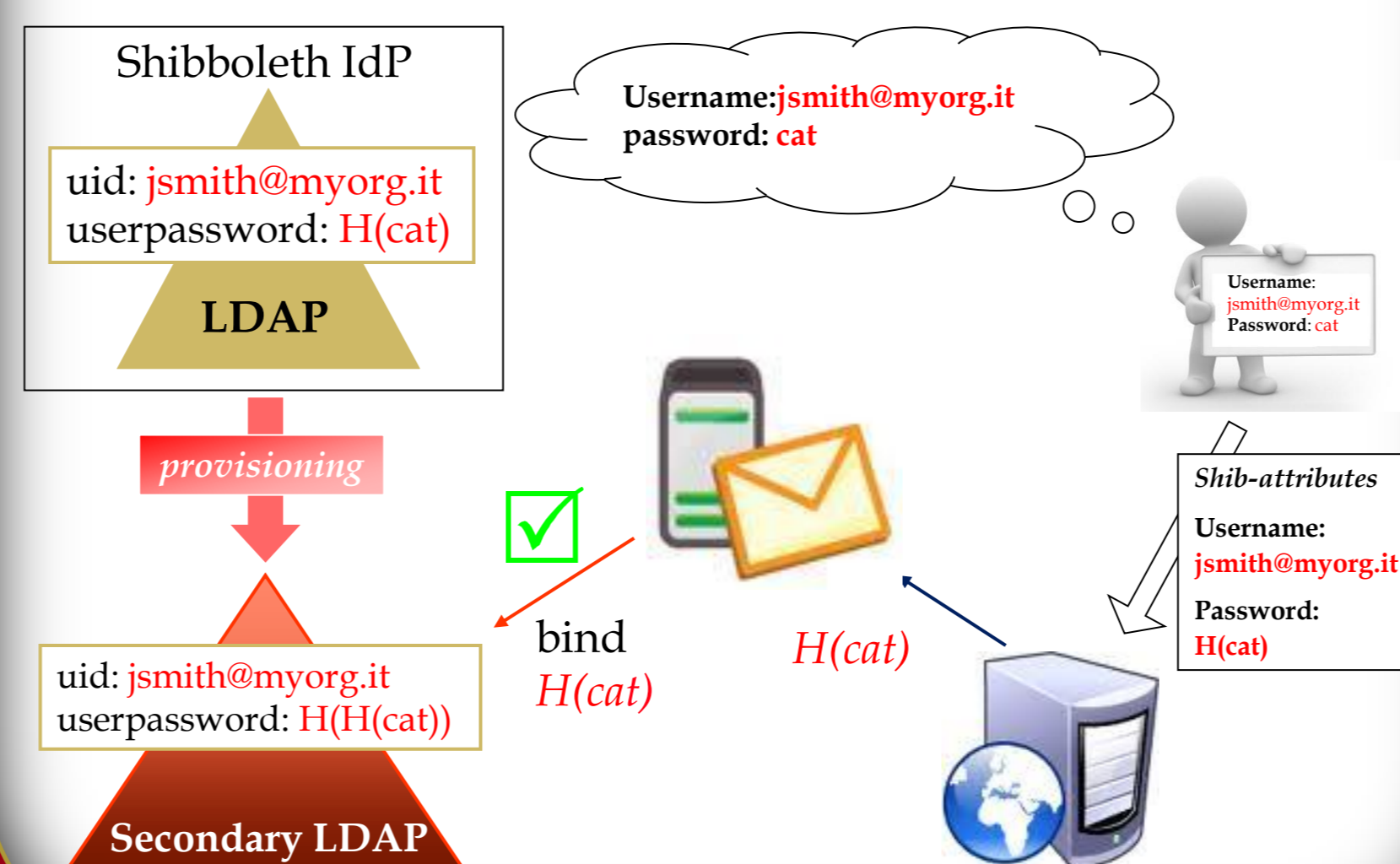
### Problem to be solved

- ❑ It's relatively easy to shibbolize the webmail server.
- ❑ The webmail server is in turn an IMAP mail client.
- ❑ How can the Mail server authenticate the user?



### Solution

A secondary LDAP with *H(H(cat))*



### Traditional solutions

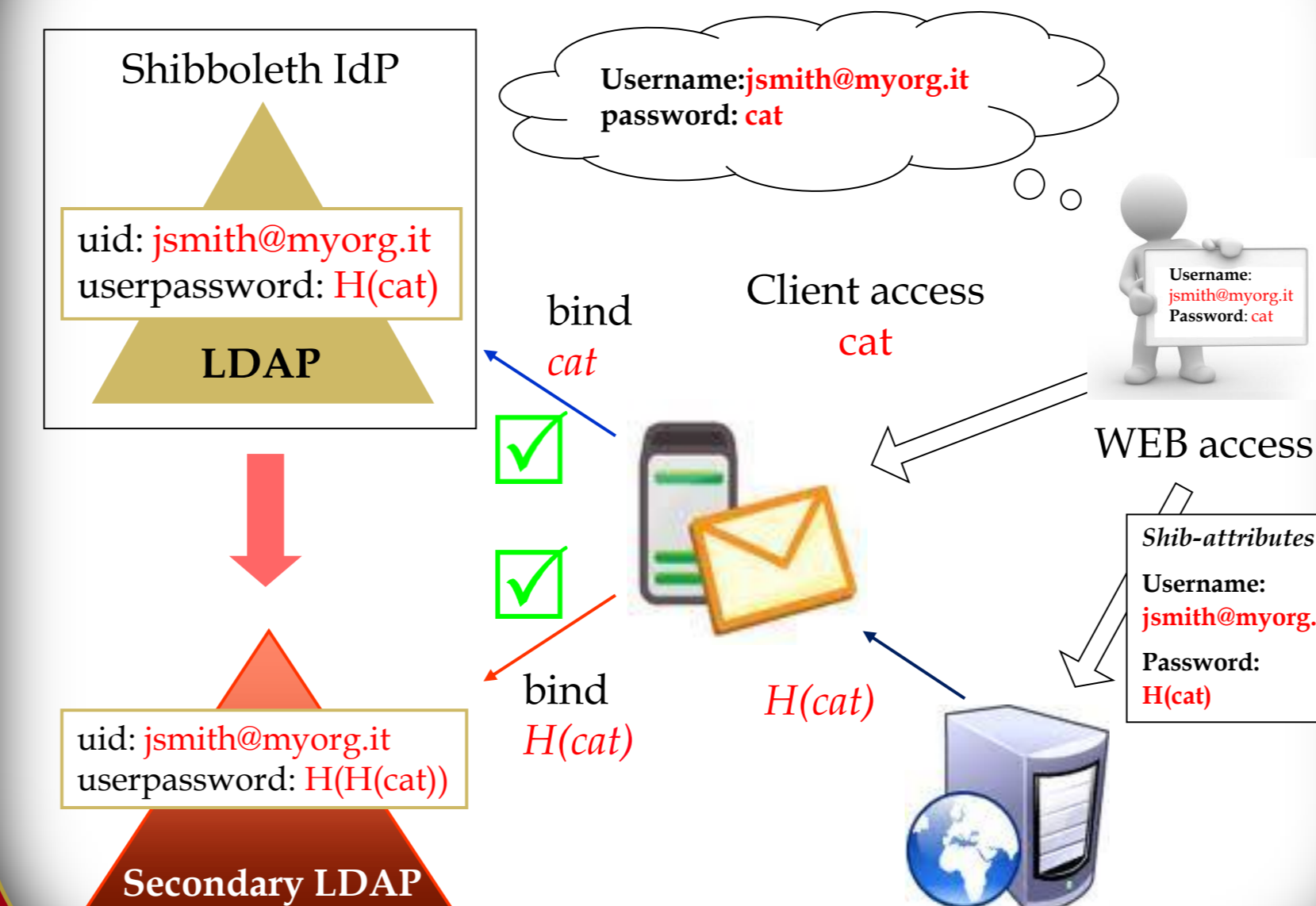
- 1) Let the webmail server gain trusted access (no credentials needed) to the mail server.

Easily done, but if the webmail server gets compromised the cracker can fetch everyone's mail!

- 2) Pass the user password to the webmail server as a *shib-attribute*. The webmail server can then use it towards the mail server.

Technically possible but highly unrecommended: the cracker could gain access to all applications under SSO

### Overview



## Conclusions

### Results

- ❑ The SSO password is not sent to the webmail server (in case the webmail server gets compromised the cracker will access only some inboxes)
- ❑ This solution is general to all 3-tier applications where the client needs also to authenticate directly at the backend server.
- ❑ This solution is easy to deploy besides existing authentication architectures with no need to change them or to interrupt production authentication services.

### Future developments

- ❑ This solution is as strong as the used hashing algorithm.
- ❑ The flexibility of the virtual directory makes it easy to set a reduced time validity for the transmitted hash, thus raising the strength of the proposed solution.