An Authentication and Authorization Architecture for the Mobile Internet

T. Braun, Th. Spreng, M.-A. Steinemann
Institute of Computer Science and Applied Mathematics
University of Bern
Neubrückstrasse 10
3012 Bern
Switzerland
{braun, spreng, steine}@iam.unibe.ch

Abstract

The paper describes an authentication and authorization architecture for mobile Internet users. The architecture is based on the Shibboleth middleware that has been developed by the Middleware Architecture Committee for Education of the Internet2 Middleware Initiative. The initial goal of this middleware was access control to digital content available in the Internet. We propose to use this middleware also for controlling Internet access via wireless local area networks. First, users need to negotiate credentials with their home organization. After a single login at their home organization roaming users can access the Internet via different (wireless) access networks. The system has been implemented on Linux platforms.

Keywords

Authentication, authorization, mobility, access control

1. An Authentication and Authorization Infrastructure for Web Services

Shibboleth is an authentication and authorization infrastructure for web resources and has been developed by the Middleware Architecture Committee for Education (MACE) of the Internet2 Middleware Initiative. A web resource can be a simple web page or a web service to be accessed via the SOAP (Simple Object Access Protocol) and HTTP (Hypertext Transfer Protocol). In each case the resource might have to be protected and should be accessible by authorized users only.
Shibboleth aims to provide a standards-based and vendor-independent web access control infrastructure, which can operate across institutional boundaries. Shibboleth allows federated user administration for resource access based on user information attributes, which are stored in databases of the users' home organizations. Shibboleth uses the Security Assertion Markup Language (SAML) for authentication and authorization message exchange. SAML has been developed by the Organization for the Advancement of Structured Information Standards (OASIS) security services

technical committee to support interoperability between entities for web access management and especially for providing single sign-on capabilities. The Shibboleth protocol runs mainly between the organization, where a user wants to access a resource, and the so-called home organization, where the user has been registered. Shibboleth supports credential transfer between the user and his home organization as well as attribute exchange between the home organization and the organization owning the resource to be accessed. Resource access can be granted based on the received user attributes. Section 2 describes the protocol in more detail.
Ideally, the Shibboleth protocol runs between the server hosting a resource to be accessed by the user and the authentication server of the user's home organization. In addition, a redirection server (called Where Are You From (WAYF) server in Shibboleth terminology) may be involved in the authentication process. However, this requires integrating the Shibboleth protocol suite into the server hosting a resource. This integration might be difficult for special software systems, in particular when the server software hosting the resource is not open source or not available. For those cases, we have developed a portal running the Shibboleth protocol on behalf of the resource. The portal needs to be adapted to management interfaces that are supported by the resource. For example, the portal might retrieve information about available services from the server via that interface and configure the server for allowing authenticated users to access the resource.

2. Authentication and Authorization for Mobile Internet Access

In this Section we propose to extend the authentication and authorization architecture for controlling access to wireless Internet services. We consider the access to the Internet via a wireless network similar to a web resource that must be protected from unauthorized access. We also assume that users belong to a certain home organization, which can authenticate its users based on the exchange of some credentials. Note that the proposed architecture is independent from the type of credentials and authentication procedures. Examples might be shared secret passwords, one-time passwords, public keys, or even biometrical data. The proposed architecture makes also use of the portal introduced in Section 1. The overall scenario is depicted in Figure 1. We call the access network through which a roaming user can access the Internet a "docking network". Internet access via docking networks should only be granted to authenticated and authorized users. The docking network is typically a wired or wireless LAN. We assume that it is publicly accessible, e.g., behind a demilitarized zone (DMZ) in an organization's network. Organizations might be universities, companies, or even Internet service providers. A firewall should be placed between the docking network and the DMZ, through which the Internet can be accessed. This firewall must be dynamically configurable. For example, a Linux based firewall with a packet filter such as iptables would be sufficient. To support ease of use, we recommend providing a DHCP service in the docking network. This service might be integrated into the firewall system.
Moreover, we need to filter the web traffic from all users that have not yet been authorized to access the Internet via the docking network. All unauthorized http(s) requests are redirected to the portal (1), which triggers the Shibboleth authentication process. The user is then redirected to the WAYF server and has to select his home

organization (2). Based on his selection he is further redirected to the authentication server of his home organization (3). There, he can use the authentication method of his home organization and authenticate himself. For the communication with the WAYF server and the authentication server, it is required that the firewall does not filter that traffic. We can achieve that by configuring appropriate filter rules in the firewall. For example, we can allow any HTTPS traffic to/from WAYF and authentication servers. This requires, however, maintaining a list with all potential WAYF and authentication servers. Once authenticated, the home organization's authentication server creates a handle, which acts as a reference to the user and sends it to the portal at the docking network (4). The portal then requests attributes about the user from the authentication server (5). The authentication server delivers the desired attributes such as user name, user group or individually subscribed services to the portal. Based on these attributes, the portal decides whether it can allow the user to access the Internet via the docking network.

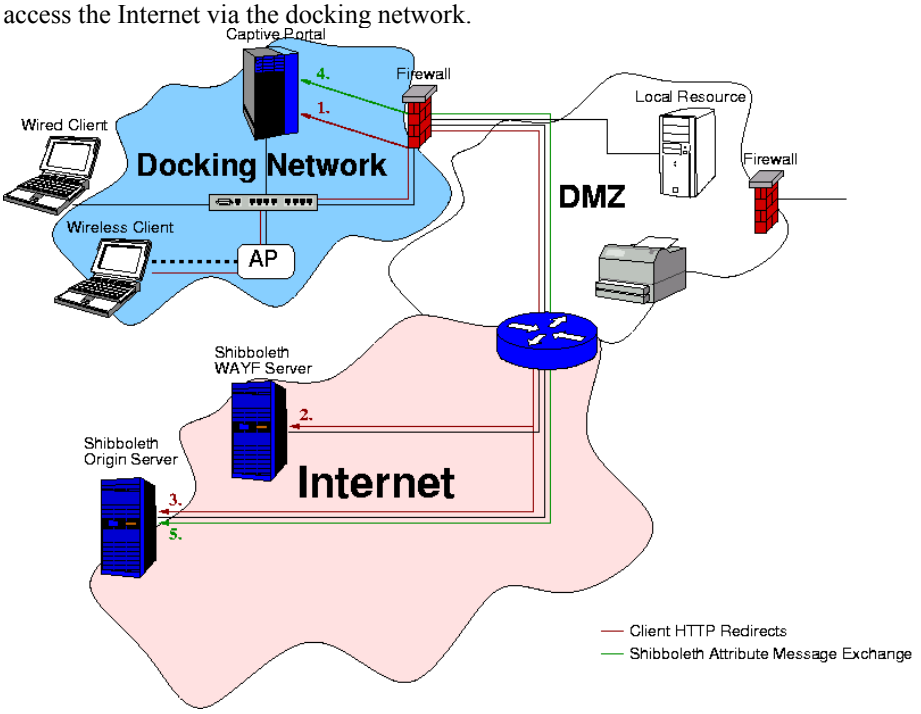

Figure 1: *Shibboleth Authentication and Authorization for Mobile Internet Access*

Note that the portal does not need a complete list of individual authorized users, but fewer entries are sufficient. For example, two organizations such as universities can negotiate to allow each other's students to access the Internet via their docking networks. In that case, the authentication server of organization A only needs to confirm that the authenticated user has already registered as a student at organization A. Configuration data must be written to the portal at the docking network of organization B before all users from organization A can access the docking network. After successful authentication, the portal instructs the firewall to set appropriate

filter rules for the device of the authenticated user, e.g. based on the IP and MAC address of the user's device. The portal should then also check periodically via ping messages that the user's device is still active. Otherwise the filter rules should be reset again.
The architecture allows that a user authenticates only once at the home authentication server, even when the user roams to various foreign networks afterwards. For each visited network the user can identify himself to the respective portal via the handle created in step 4, which can be used by the portal to request attributes from the home organization server. The main concerns of the architecture are the limited possibilities to prevent MAC and IP address spoofing in the docking network. In order to handle this issue, advanced security functions in IEEE 802 LANs are required. Another issue is that the components from the different organizations involved in this architecture such as the authentication server and the portal belong to the same federation, which defines a trust relationship. However, the different entities have to authenticate their messages exchanged during the Shibboleth protocol. We propose to use public key infrastructures for authentication among these entities.

3. Implementation

The implementation has been done on a single server which acts as a portal, network access gateway, firewall, and DHCP server simultaneously. Currently, the server is running a Debian Linux distribution, but the software could also be easily adapted to other Unix-based operating systems. As soon as a new user accesses the docking network all unauthorized HTTP and HTTPS traffic is redirected to the network access gateway running an Apache web server. The user is presented a welcome screen and a link to log in using Shibboleth in order to get network access. This link leads to a local Shibboleth-protected part of the web server and thus requires the user to authenticate at his home organization. Once the user has been successfully authenticated, the authorization module written in PHP decides based on the supplied attributes whether network access should be granted or not. If the user is allowed to proceed, the PHP code in the user front-end scripts will call an access control script written in Perl, which will trigger the firewall to insert the appropriate rules. This script is very simple and only acts as a wrapper for inserting and deleting firewall rules. After this process the user is redirected to a second page that contains information about the authorization status. When this process has been successful, the user may access the Internet through the access gateway.
All active sessions will be monitored by a network activity daemon written in Perl. The purpose of this process is to check whether active users are still using network access. If this is not the case it will call the access control script to log out inactive users. There are several checks that can be done to decide whether a user is still active or not, currently only ping checks and traffic measurements are taken into account but new tests can be easily added if needed. The web server also hosts an administration front-end which can be used to retrieve information about current users, session times and so on. In addition, administrators can define Shibboleth user attributes required for granting network access.

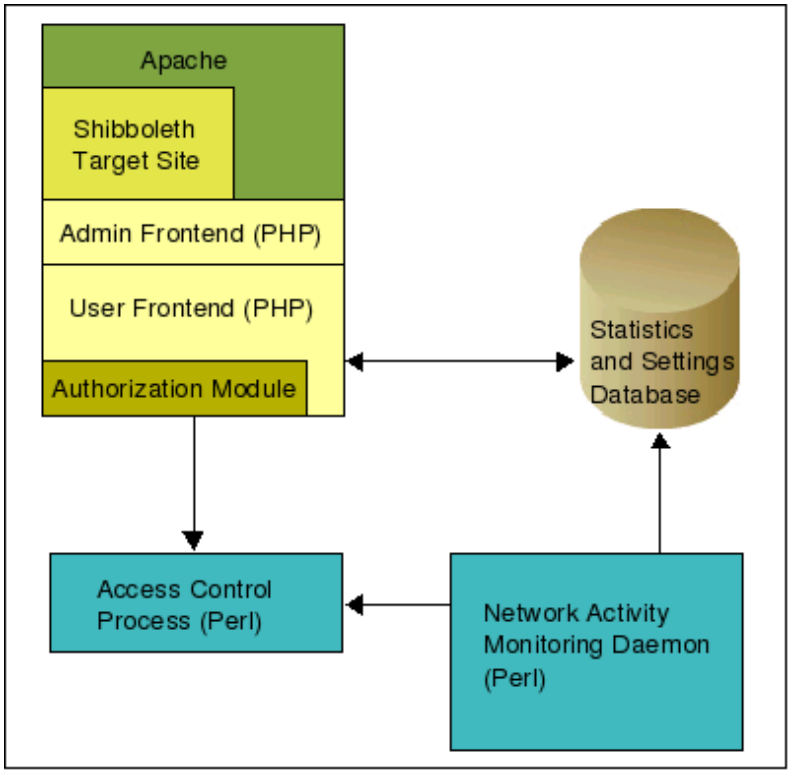

Figure 2: Software Components

4. Conclusions and Outlook

We have described a single login architecture to support mobile users roaming among different wireless Internet access providers. The architecture allows efficient and scalable user and access management. With our current solution, a user has to run a HTTPS capable web browser to get wireless access to the Internet. Future research might address solutions that do not require running a web browser. In particular for smaller devices such as personal digital assistants such a solution might be useful.

5. References

[1] R. L. Morgan, S. Cantor, S. Carmody, W. Hoehn, K. Klingenstein: "Federated Security: The Shibboleth Approach", EDUCAUSE Quarterly, Vol. 27, No. 4, 2004

[2] M.-A. Steinemann, T. Braun: "A generic broker portal linking authentication and authorization infrastructures and resources", *European Journal of Open and Distance Learning (EURODL)*, October 10, 2004, ISSN 1027-5207.