13.3 Known Partners

An organization conducting e-commerce, offering services, or publishing information using Web technologies must make their service easily accessible to their users. However, if these interactions are commercially sensitive or of a high value, we want to ensure that the users with whom we are interacting are who we think they are, and the users themselves want to be sure that our system is what they think it is. By introducing a system of KNOWN PARTNERS (442), identified uniquely in a way that can be authenticated, we can be sure of who is interacting with our system. We can also prove to users that we are who they think we are.

Example

A commercial Internet system offers two Web-technology interfaces: one for the general public and the other for business partners. The business partner interface allows the users to place orders for goods, often with a value that runs to many tens of thousands of dollars. Once the order is placed with the Web-technology system, it is sent to the corporate ordering facility. This initiates a number of supply-chain-management functions, culminating in the goods being shipped to the business partner along with an invoice for the goods.

If we allowed anyone to access this system anonymously, we would run the risk that, either maliciously or accidentally, orders would be placed by users not authorized to do so. This could result in goods being shipped in error, invoices being issued incorrectly, and business partners claiming that orders shipped to them were never placed by them.

Equally, users will be less willing to use the system and to submit information such as credit details and user information for an order, if there is a chance that someone is 'spoofing' the system, for example offering something that looks like our system, but is in fact an operation set up to collect information that can be used to commit fraud.

Context

An APPLICATION SERVER ARCHITECTURE [Dys04] has been adopted to deliver an Internet technology application. The business logic and dynamic Web content generation of the application resides on application servers, while all static content is provided by Web servers that also act as reverse proxies (see PROTECTION REVERSE PROXY (457), INTEGRATION REVERSE PROXY (465), and FRONT DOOR (473)) for the dynamic Web content. The application provides commercially-sensitive or high value services to a restricted set of users.

Problem

We want to provide a system that allows us to collaborate with an organization either as a customer or as a business partner. How can we validate the identity of an organization so that we can be sure they are who we think they are, and they can be sure that we are who we say we are?

Solving this problem requires you to resolve the following forces:

- We want to make the system as easy to access as possible to encourage business: this is probably one of the reasons we chose to offer the system via Web technologies in the first place. However, we need to balance accessibility against the need to identify and authenticate users, and to protect users from anyone who is trying to spoof our system.
- Lightweight security mechanisms such as user-name and password combinations are typically one-way: they identify the user to the system, but not vice-versa. We could adopt a lightweight approach, but these types of mechanisms are relatively easy to break, and the user is often required to provide information that is valuable to anyone that has gone to the trouble of setting up a spoof system.
- The cost of an extensive security solution will be high, but the cost of invalid system use may also be high in terms of theft and loss of customer confidence. If the potential rewards from the attack are high in terms of financial gain or publicity, the risk of such an attack will be higher. The scope, and hence cost, of any countermeasure must be commensurate with the level of perceived threat and the potential cost of the fraud.

Solution

Ensure that access to system functionality and data is restricted to known partners who must authenticate themselves in a secure manner. This 'secure manner' should involve some form of two-way exchange such that the user is identified to the system and the system is shown to be what the user thinks it is. In effect, the user and the system are both identifying each other as KNOWN PARTNERS (442) with whom they want to interact.

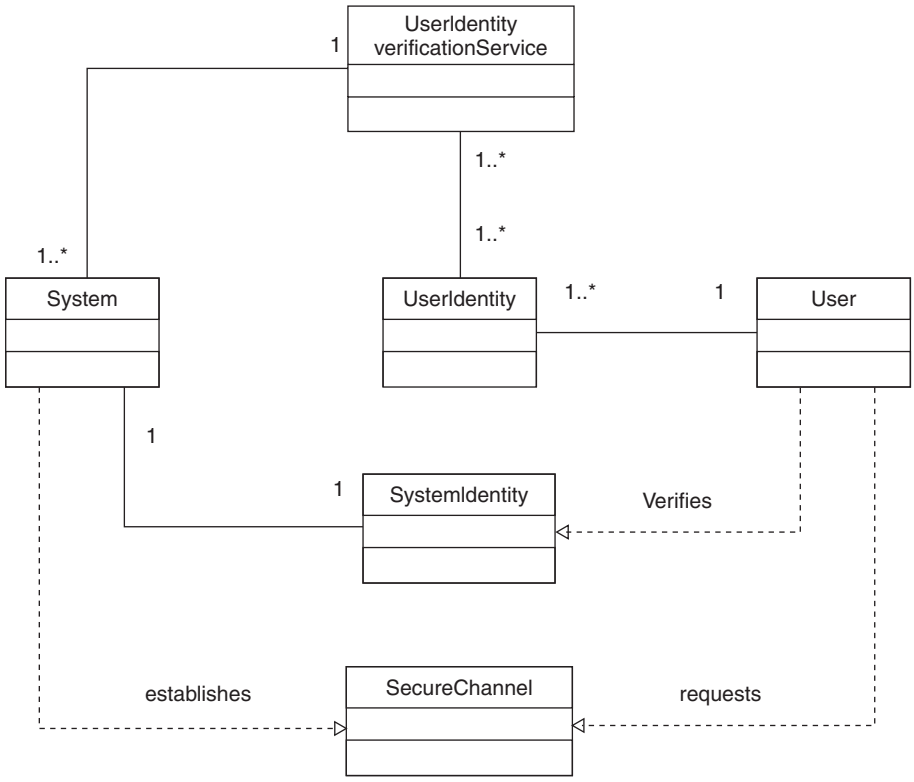
Structure

This pattern requires the following elements:

- System identity. The system has an identity that verifies to the user that the system is what they think it is.
- User identity. The user has an identity that verifies to the system that the user is who it thinks it is. This identity can be passed through the system to provide

non-repudiation of interaction: that is, the user cannot claim that an interaction was performed by someone else, and that they should not be responsible for the consequences of the interaction, because the interaction is effectively 'signed' with their identity.

- User identity verification service, a service either provided by the system or by a trusted external agency that verifies that any user identity submitted to the system is valid.
- SECURE CHANNELS (434)—identities are usually exchanged via a secure channel, as well as any further interactions between the user and the system.

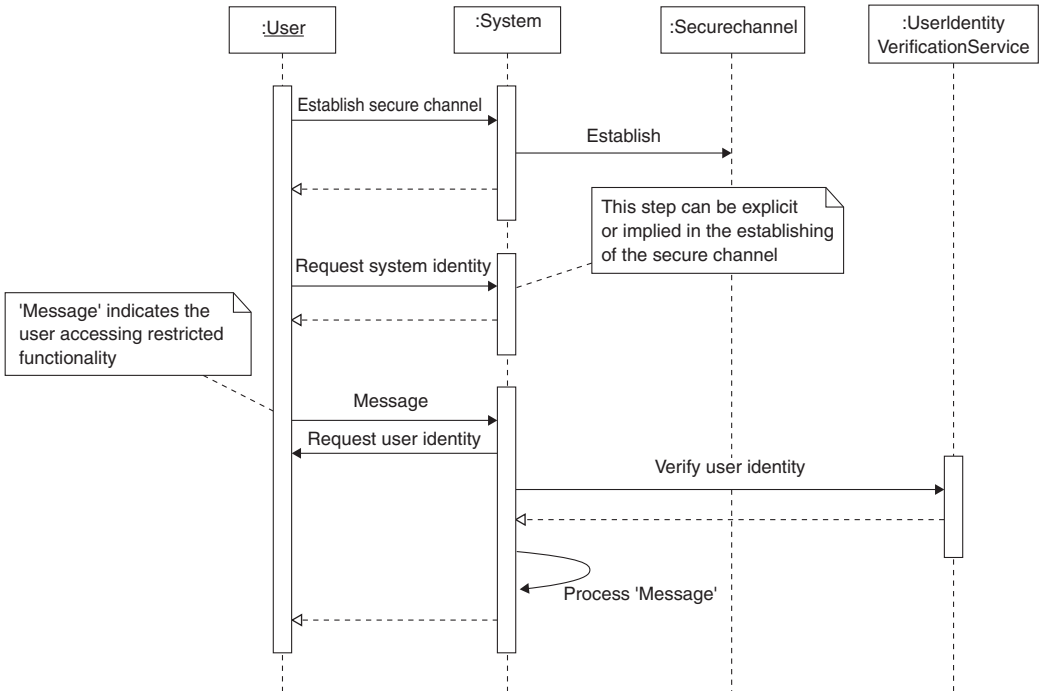


The structure of KNOWN PARTNERS

Dynamics

The first scenario shows a successful interaction between the user and system. The user wants to send the message `Message` to the system which requires access to restricted

functionality. First, both the `User` and the `System` need to establish that they are each KNOWN PARTNERS (442).



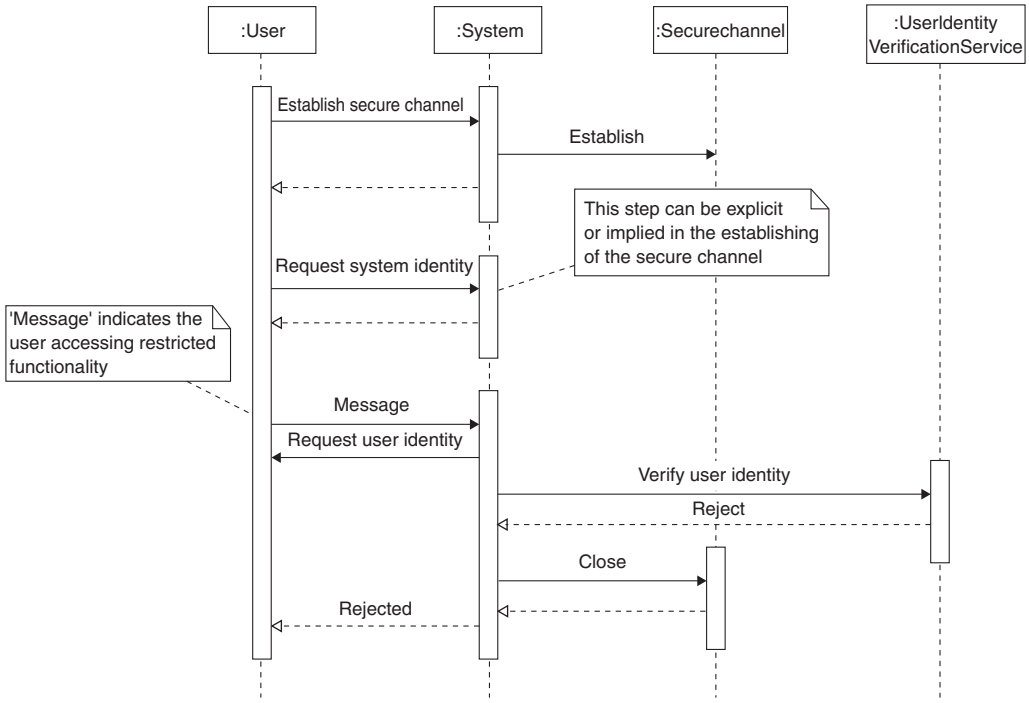
Successful interaction between KNOWN PARTNERS

The second scenario shows an invalid client identity being detected and blocked by the system. See figure on page 446.

Implementation

One of the commonest implementations of KNOWN PARTNERS (442) is to use digital certificates for both the system and client identities. In this case the system provider obtains a certificate from a known certification authority (CA). This certificate has the domain name of the system embedded in it. When the user first connects to the system, the system provides the certificate, while the browser verifies that it is correct for the domain name and has been authorized by a known CA. This prevents spoofing, as a spoof organization should not be able to obtain a CA-authorized certificate and, if they do, it will not be tied to the domain name of the system provider.

To access any restricted functionality, the user also needs to obtain a CA-authorized certificate. This certificate is installed directly into the browser, where it is secure from



Rejecting an invalid client

tampering—although the machine on which browser is installed may not itself be secure. When the user's browser has verified that the system provider's certificate is valid, the user must then provide their certificate to the system. The user identity verification service then checks that the user certificate is also valid—that is, it is CA-authorized and has not been revoked or expired by the CA. If the certificate is valid, the user is given access to the restricted functionality.

To ensure non-repudiation, it is not uncommon for the system to require that the user certificate is passed to the system for every interaction or culmination of interaction, such as the confirmation of order placement. This means that the certificate's details can be stored with the results of the interaction (or passed to back-end systems). As long as the system provider can demonstrate that there is no way within the system for one user's certificate details to be replaced by another, it is very hard for the user to contend that the interaction was not carried out by them, and that they therefore should not be liable for its consequences.

How the client obtains their certificate is dictated by the level of security required by the system provider. One option is for the system provider to act as their own CA: they provide the certificate to the user and maintain the set of valid user certificates. Another option is to partner with a recognized CA and outsource the verification of

user identity, issuing of certificates, and maintenance of the revocation list to them. CAs will offer different levels of user identity verification, from a simple check of on-line identity through to a face-to-face identity verification.

Example Resolved

The commercial organization implements a certificate-based KNOWN PARTNERS (442) mechanism. It obtains a certificate from a recognized CA which it uses to set up an SSL-based SECURE CHANNELS (434). All access to restricted functionality must take place over that SECURE CHANNELS (434).

The organization decides to act as its own CA because it already has a lot of face-to-face interaction with its business partners. Each business partner that requires access to the on-line functionality is issued an individual certificate signed by the organization. When the user accesses the restricted functionality, they are required to provide the certificate, which the system then checks against its own revocation list.

At the culmination of an interaction such as the confirmation of order placement, the individual user ID embedded in the certificate is passed with the order details to the corporate ordering facility.

Known Uses

KNOWN PARTNERS (442) mechanisms are becoming increasingly common for commercially sensitive or high-value online interactions. The authors have worked with several companies that implement a certificate-based KNOWN PARTNERS (442) scheme to provide access to 'extranet systems' as well as internal resources such as document and code repositories. The UK government also uses a certificate-based scheme for its 'government gateway' (`http://www.gateway.gov.uk/`), which provides access to functionality such as on-line filing of business tax returns.

Variants

Multi-part user identity. The use of digital certificates actually ties the interaction to a browser on a machine rather than to an individual user. This is advantageous if we want to allow multiple users to act on behalf of a business partner and we don't care which individual, but is a liability if we want to identify individual users. A common variant of certificate-based user identification is the addition of a password or PIN individual to each user, that must be supplied at the same time as the certificate. Multi-part user identities are also useful in the case of machine theft, as possession of the certificate alone is not sufficient to access the restricted functionality of the system.

Hardware token. Rather than using certificates for user identification, a hardware 'token' is issued to each user. The token usually provides a key that changes frequently

and must be provided to the system on log in—either the key is displayed and the user types it in, or the hardware token is physically connected to the machine and provides the key automatically. Hardware-token based systems also frequently use a multi-part user identity, as theft of the token is usually easier than theft of the client machine, and less readily noticed by the user.

Consequences

The following benefits may be expected from applying this pattern:

- Security is improved, because the system can be sure that any user accessing the system is who it thinks they are.
- User confidence is improved, because they can be sure they are not accessing a 'spoof' system.

The following potential liabilities may arise from applying this pattern:

- Performance is slightly impacted, because exchanging and verifying system and user identities introduces overhead in processing a user's request.
- Availability is potentially impacted, because the user identity verification service becomes a single point of failure for access to restricted functionality.
- Manageability is impacted, because system and user identities must be actively managed to maintain the required level of security.
- KNOWN PARTNERS (442) is significantly more expensive to implement and maintain than a lightweight mechanism based on passwords.