Single Sign-On across Web Services

Ernest Artiaga

CERN - OpenLab Security Workshop – April 2004

Outline

- Motivation and goals
- Tools
- Single sign-on
  - Impersonation: Mapping certificates to accounts
  - Providing certificates to users
  - Issues and actual status
- Summary and conclusions

Motivation

- The environment:
  - Services offered through web
    - Applications using web servers as user interface
  - Clients on both Windows and Unix platforms

- What we want (*and what the users ask for*):
  - Authentication mechanism valid across platforms
  - Single sign-on

Goal

- Letting users access authorized resources…
  - Restricted web pages
  - Web-based services (mail, …)
- …without re-typing usernames and passwords (single sign-on)

Tools

- Two different technologies
  - Kerberos
    - Well-known for certain applications
    - “Supported” by modern operating systems
  - PKI/Certificates
    - Widely spread
    - Portability across platforms

Tools

- The drawbacks…
  - Kerberos
    - Incompatible extensions
    - Few “kerberized” applications
- So, we decided to try PKI/Certificates as a base for a Single Sign-On mechanism.

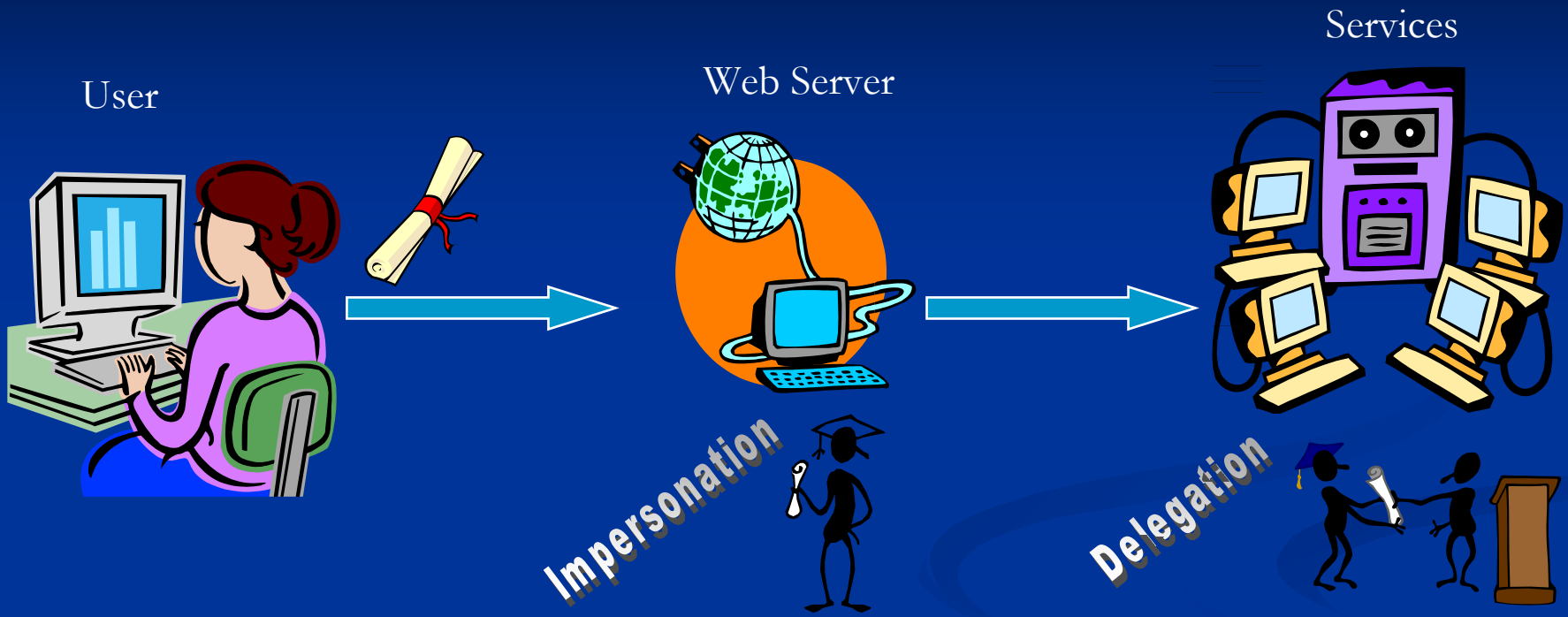
Single Sign-on

- CERN users have accounts in both Unix and Windows environments
- Services are not replicated in both systems
- Logon and Authentication mechanisms are different
  - A user must type his/her credentials again and again
  - Can the PKI/Certificates help?

Single Sign-on: basic web access

- PKI/Certificates can be used to protect access to web pages
- They provide portable authentication and access control
  - Available for both Apache and IIS servers
- … But this is mainly local access
  - What happens if the server needs to access remote data?

Single sign-on



- We must provide the user with a valid PKI/Certificate
- We must trust the web server
  - It will impersonate the user!

Impersonation in IIS

- Based on the Windows Identity Mapping mechanism
  - Maps a certificate to a specific account
- The identity mapping can be managed at two different places:
  - The IIS server itself
  - The Active Directory

IIS mapping

- Specific to a web site
- Flexible many-to-one mapping rules
  - Based on issuer and subject of the certificate
- Provides a ticket valid for delegation
  - I.e. remote resources can be accessed
- Username and password must be provided when setting the mapping
  - but they are not kept synchronized with windows accounts!

AD mapping

- Common for all web sites in the domain
- Limited many-to-one mapping
  - There is a single account for all the certificates coming from the same issuer CA
- One-to-one mapping is the most convenient
- Provides a ticket valid for delegation since Windows .NET Server/IIS 6.0

AD mapping (II)

- Two flavors: manual and automatic
  - In manual mapping, the administrator must specify which certificate maps into which account (can be done programmatically)
  - In automatic mapping, the certificate must contain an extension (subjectAltName), with the User Principal Name (UPN) of the account in the otherName field
    - No explicit mapping is needed
    - Originally designed for smart cards

Impersonation in Apache

- Impersonation via Kerberos ticket
- Uses extra software: Kerberos leveraged PKI
  - KCT (Kerberos Certificate Translation)
  - Mod_KCT (Apache module)
- Procedure:
  - The user sends a PKI/Certificate (obtained through the KCA) to Apache
  - Apache uses KCT to recover the user's Kerberos ticket
  - Apache uses the ticket to access user's remote resources

Providing certificates to users

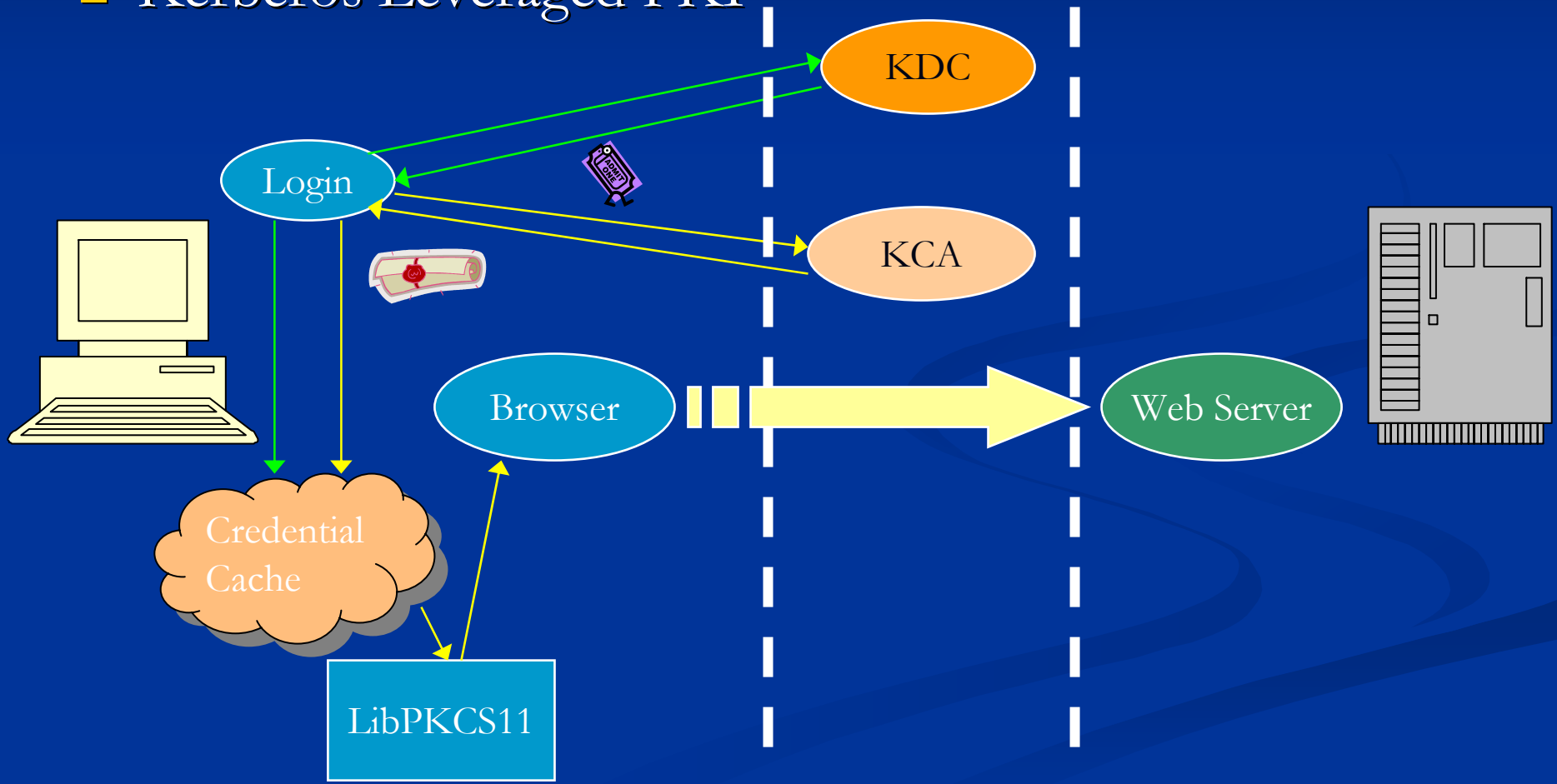
- *There is a risk of users not taking care of their certificates…*

- It should be a transparent mechanism
- It should be easy
- It should be secure

- Both Unix and Windows users receive a Kerberos ticket during logon
  - We can issue a PKI/Certificate for a Kerberos ticket

Providing certificates to Users

- Kerberos Leveraged PKI



Providing certificates to users

- KCA (Kerberized CA) supports Kerberos V (Windows 2000 compatible)
- KCA clients are available for Unix and Windows
- PKCS11 library (smart card emulation) is also available for Unix and Windows
- We have short term certificates

Issues: certificate restrictions

- The user certificate must contain a series of extensions properly filled and encoded, so that the web server accepts it and maps it to the right account.
  - subjectAltName
  - cRLDistributionPoint
  - keyUsage
  - extendedKeyUsage
  - Expiration date properly set
- Possible CAs:
  - Microsoft recommends MS Enterprise CA
  - Entrust CA also works
  - … We used OpenSSL… ☺

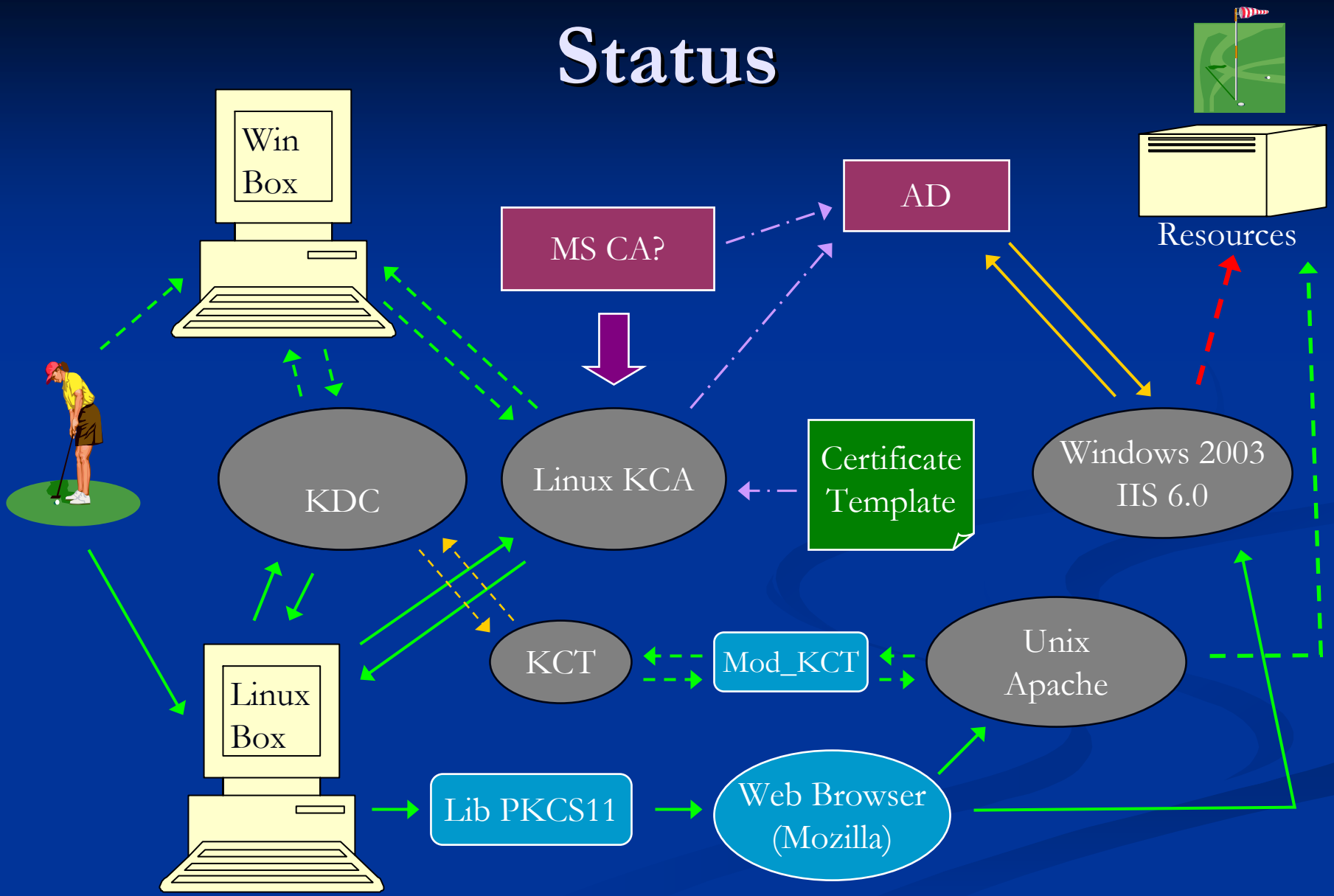
Issues: server side CA restrictions

- It is *possible* to use a non-MS CA with an IIS server, but…
  - … it should behave as Microsoft's one
- The CA certificate must be added to the NTAuth store in the registry… manually.
- It should create the same AD entries and fill them properly
- Certificates and CRLs must be published in the AD

Issues: web applications

- Lack of integration between the authentication mechanisms for the web servers and the applications behind them
  - First, authenticate with the web server…
  - Then, authenticate again with the application!
  - E.g. some web mail applications…
- Despite the necessary security infrastructure being there, some applications keep
  - Using their own security mechanisms
  - … or using it only "*internally*".

Status


Win Box
MS CA?
AD
Resources
KDC
Linux KCA
Certificate Template
Windows 2003 IIS 6.0
KCT
Mod_KCT
Unix Apache
Linux Box
Lib PKCS11
Web Browser (Mozilla)

Summary and conclusions

- In theory, it is possible to achieve cross-platform single sign-on
- But full functionality has issues…
  - Lots of components involved (KDC, KCA, AD…)
  - Compatibility (not fully documented requirements)
  - Intrinsic limitations
    - Extensions not present in the KCA certificates
    - Integration between applications and servers

Questions?