Using Kerberos for Web Authentication

Wesley Craig
University of Michigan
Outline

- Basic Auth
- WebSSO
- SASL & HTTP
- Kerberos & TLS
- SPNEGO
- PKI, PKI, PKI

For each technology, a brief overview, drawbacks, and benefits. All informed by our work for University of Michigan on CoSign.
Proxy Authentication

Three sorts of proxy: web server enforces authZ; web server uses, e.g. SASL, to authN as itself authZ as user, application enforces authZ; web server authN as user for applications that don’t support split authN/Z, e.g., AFS.
Basic Auth

- Defined in RFC 2617
- Most browsers & web servers implement it
mod_auth_kerb Basic

Browser: GET
Server: 401
WWW-Authenticate: Basic realm="some text"
Browser GET
Authorization: Basic base64(user:password)
mod_auth_kerb Basic

Browser ➔ Web Server ➔ KDC

Browser: GET
Server: 401
WWW-Authenticate: Basic realm="some text"
Browser GET
Authorization: Basic base64(user:password)
Basic Auth Risks

- User gives password to every web server
  Breaks single sign-on
  Trains users to freely give their password
  Is the server secure?
- Every web server needs SSL (or not)
- Every web server needs a keytab (or not)

Should I be sending my password to this server? Has this server been compromised?
Basic Auth Benefits

- Widely supported & well understood
- Works with WebDAV
WebSSO

- Typically “Form & Cookie”
- Typically only single sign-on for web services
- Examples: CoSign, WebAuth, CAS, etc.

Typically, because WebSSO’s can also leverage “true” SSO
CoSign Design

- Compartmentalized Security
- Kerberos V
- Proxy Kerberos Tickets
  http://filedrawers.org
- High Availability
- Global Logout
CoSign Extensions

- Centralized Guest Accounts
- Proxy CoSign Cookies
- Re-Authentication
- Multi-Factor
- Apache Authentication Modules
- X.509
CoSign Risks

- Requires cookies, which can be stolen
- May use passwords, which can be stolen
CoSign Benefits

• Broad browser support
• Can leverage: Basic Auth, SPNEGO, Shib, PKI, other WebSSOs, etc.
• Simple for users to understand
• Simple for CoSign-protected services
Kerberos over HTTP

- Kerberos over TLS (aka SSL)
  RFC 2712
  lynx, curl, stunnel
- SASL over HTTP
  draft-nystrom-http-sasl-12.txt (expired)

No support for graphical browsers.
SPNEGO

- Defined in RFC 4178
- Simple and Protected Generic Security Service Application Program Interface Negotiation Mechanism

HTTP Negotiate: draft-brezak-spnego-http-05.txt
HTTP Negotiate: draft-jaganathan-kerberos-http-01.txt (expired in January)
mod_auth_ldap SPNEGO
mod_auth_kerb SPNEGO
SPNEGO Risks

- Limited browser support and/or complex configuration
- Web server support
- Kerberos client support

Browsers don’t necessarily behave as expected or in a friendly way. Some don’t support delegation. Supporting “Kerberos” might mean supporting AD on some platforms.
SPNEGO Benefits

- True SSO
- “Delegation” works for tiered/proxied services
- Active community
SSL Client Authentication

- Distribute X.509 client certificates to users
- What about Kerberos?
PKI - client certificates

Web server sends transcript of SSL handshake to credential translator and gets back kerberos credentials.
Web server sends transcript of SSL handshake to credential translator and gets back Kerberos credentials.
PKI - pkinit & SPNEGO
PKI - pkinit & SPNEGO

Browser → Web Server → KDC
PKI - junk certificates

Diagram:
- Browser
- Web Server
- KDC KCT
- KCA

Connections:
- Browser to Web Server
- Web Server to KDC KCT
- KDC KCT to KCA
PKI - junk certificates

Diagram:
- Browser
- Web Server
- KDC
- KCT
- KCA

Connections:
- Browser to Web Server
- Web Server to KDC
- KDC to KCT
- KCT to KCA
- KCA to Browser
PKI - junk certificates

Browser ➔ Web Server ➔ KDC KCT

Browser ➔ KCA
PKI - junk certificates

- Browser
- Web Server
- KCA
- KDC
- KCT
SSL Client Authentication Risks

- Need either PKI or client software
- Hard for users to understand
- Certificates can be stolen
- More complex solutions inherit all the problems of their underlying components
- Not widely adopted
SSL Client Authentication

Benefits

• True SSO
• PKI is useful outside of browsers
• PKI is useful beyond authentication
University of Michigan

- Deploy Multi-Factor AuthN in CoSign
- Deploy Client Certificates in CoSign
- Deploy SPNEGO in CoSign
- Deploy WebDAV with Basic Auth
Q & A

http://weblogin.org

cosign@umich.edu

wes@umich.edu