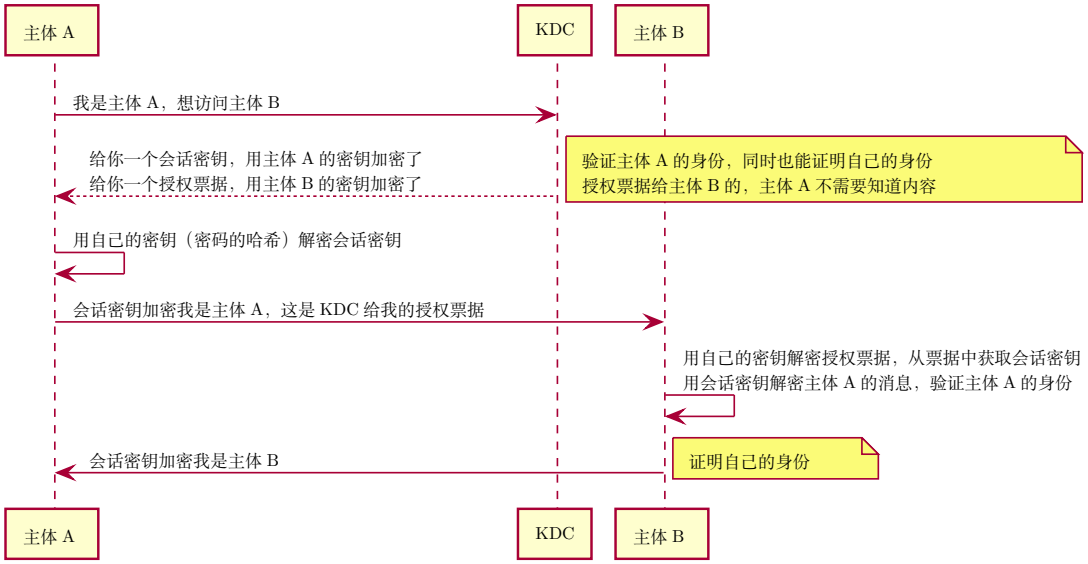
目录

1 Kerberos、GSSAPI 和 SPNEGO

GSSAPI 是一种安全机制，定义了通用的安全接口规范。应用可以使用 GSSAPI 创建安全的上下文进行身份验证，而不用关心安全的实现细节。

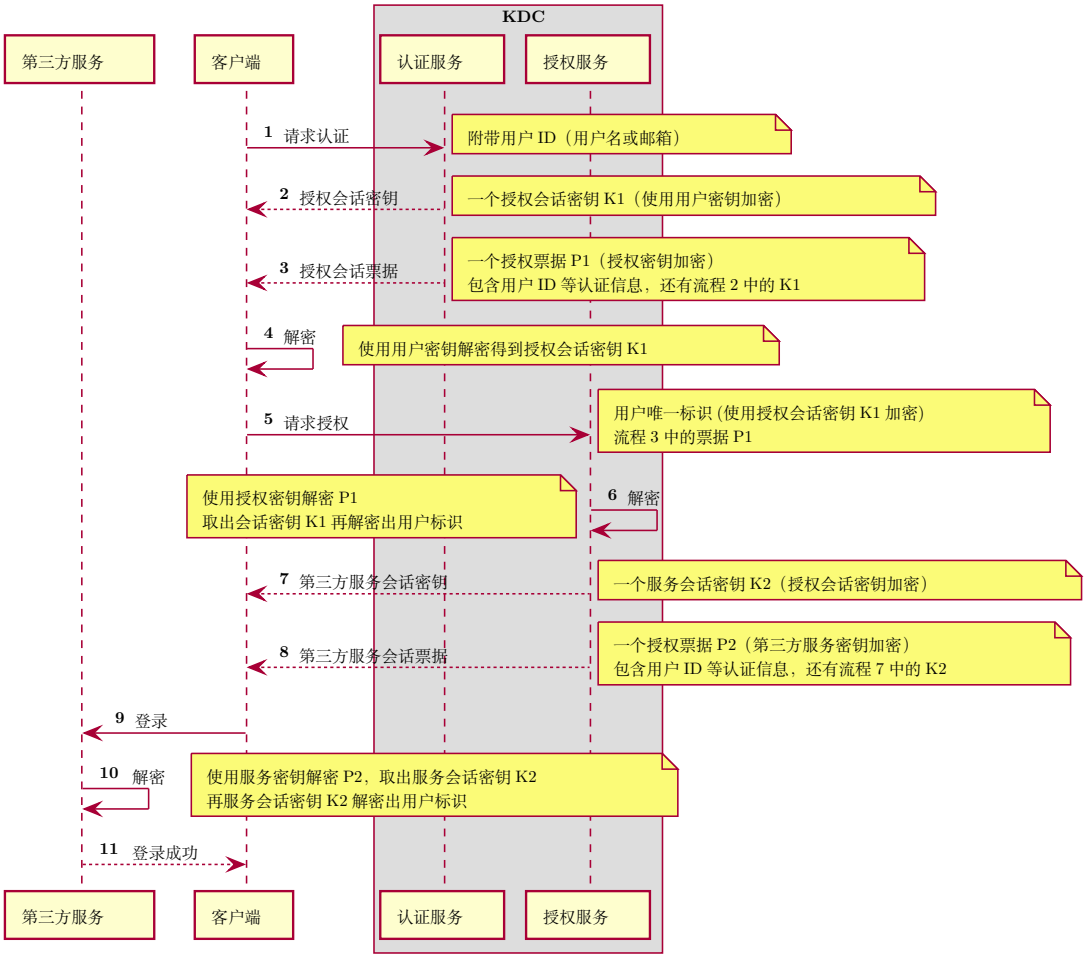
Kerberos 是一个协议，并实现了 GSSAPI 的接口，也代指其具体实现服务程序 MIT Kerberos(krb5)。
支持的应用有 ssh、pam、nfs、curl、postgresql、mysql、hadoop、chrome、firfox、nginx、ad 域等
Kerberos 协议允许使用对称密钥加密在非安全的传输层进行双向身份验证，kerberos 需要一个可信的第三方密钥分发中心（KDC）。



SPNEGO 是一种认证协商的协议，用于自动选择认证方式并简化了认证流程，通常选择项是 Kerberos 和 NTLM，可在浏览器中基于 HTTP 协议进行验证。

2 kerberos 协议

KDC 包含认证服务和授权服务，他们知道自己的密钥，也知道其他主体的密钥，其他主体只需要知道自己的密钥。



经过授权认证后，客户端信任第三方服务，第三方服务也信任客户端。使用对称加密存在密码爆破的问题，认证授权过程中加入临时生成的会话密钥减少被攻击的可能性，会话密钥有一段有效期，也能提升性能。

3 kerberos 优点和缺点

- 密钥不会被窃听
  密钥不经过网络传播，票据对应到每个主机，内部使用哈希加盐的方式存储密钥。
- 配置复杂
  服务端配置较为复杂，客户端也需要进行配置。
- 单点登录
  只需输入密码验证一次，任何使用 geeapi 的应用都可以自动登录。
- 单点故障
  中心话的方式导致必须保证可用性，否则所有相关应用都会收到影响
- 兼容性广泛
  可应用于各个基础设施，甚至是操作系统，而不仅仅是 web。
- 很旧的协议
  九十年代的协议，广泛的应用，意味着更大的攻击范围

4 Kerberos 常用操作

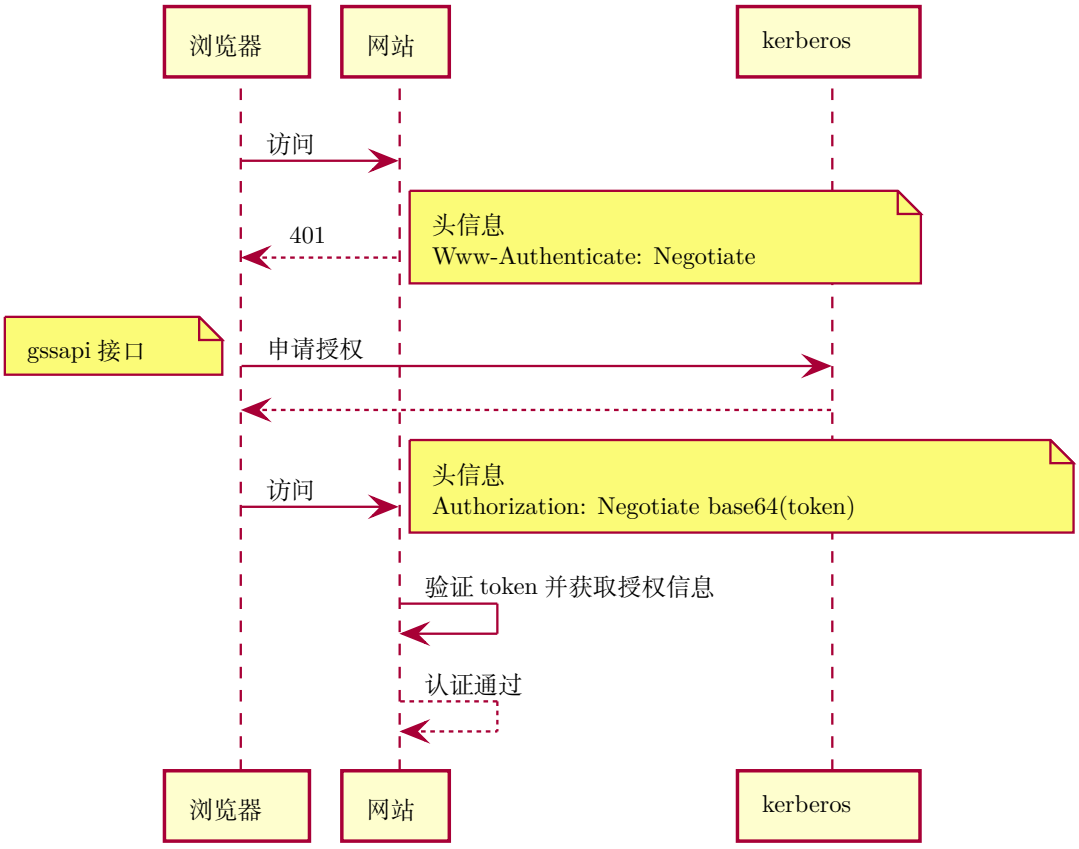
```
# 登录
kinit admin/admin@EXAMPLE.COM
# 查看票据
klist
# 登出
kdestroy
# 进入管理员界面
kadmin
    # 创建主体（使用随机密码）
    addprinc -randkey HTTP/b.example.com@EXAMPLE.COM
    # 查看主体列表
    listprincs
    # 导出到密钥表
    ktadd -k b.keytab HTTP/b.example.com@EXAMPLE.COM
```

5 SPNEGO 编程

SPNEGO 用于 C/S 在未知对方支持的身份认证协议的情况下，协商认证协议使用的，有微软制定，得到所有主流浏览器支持，安全认证协议一般选项有 NTLM 和 Kerberos，NTLM 因为加密算法太弱，已不推荐使用。
SPNEGO 使用 gssapi 接口进行安全认证，并通过 http 头传递认证 token。
网站对接 SPNEGO 不需要和 KDC 通信，但需要配置从 KDC 导出的密钥，用于解密授权 token。
SPNEGO 提供单点登录但不提供传输加密，需使用 https 保护连接。

5.1 使用范例

```
curl --negotiate -vv -u : http://b.example.com:8080
google-chrome-stable --auth-server-whitelist="*example.com"
```



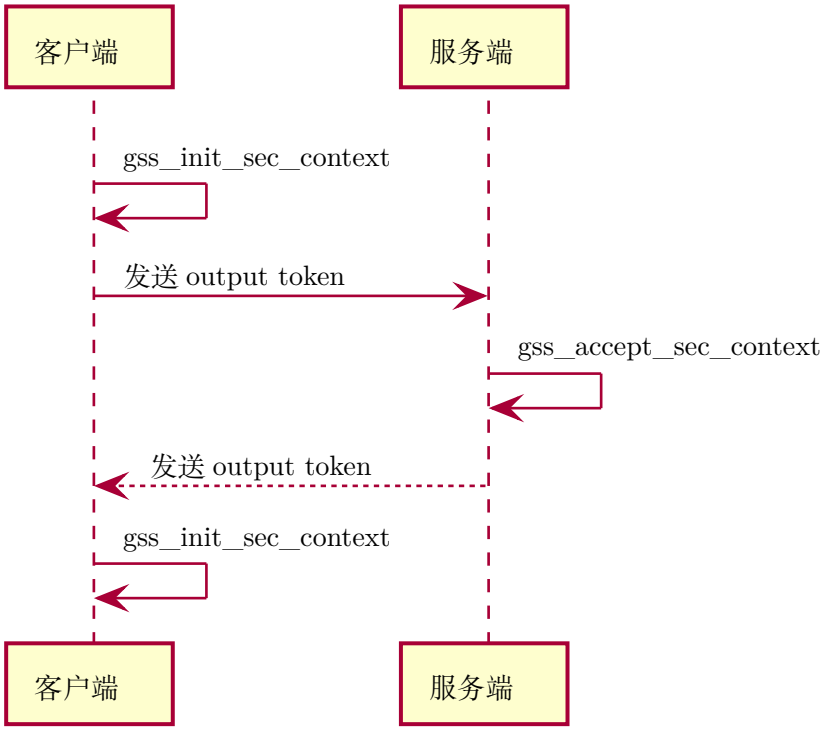
6 GSSAPI 编程

6.1 GSSAPI 客户端开发

- 初始环境，配置 krb5 user
- 使用 kinit 命令进行登录
- 使用 gss_import_name 函数导入服务名
- 使用 gss_init_sec_context 初始化上下文，获取 output token
- 将 output token 发送到服务端，并读取返回的 input token
- 再次执行 gss_init_sec_context 并传入 input token

6.2 GSSAPI 客户端开发

- 配置密钥路径到环境遍历
- 接收客户端发送的 input token
- 使用 gss_accept_sec_context 初始化上下文，并传入 input token
- 将 output token 发送到客户端



7 参考链接

GSSAPI 规范
SPNEGO 规范
Kerberos 规范
GSSAPI 编程示例 (c 语言)
GSSAPI 编程示例 (cgo)
SPNEGO 编程示例 (golang)