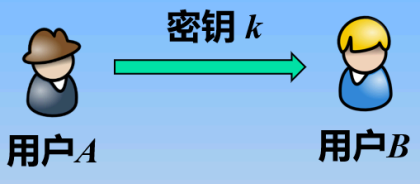
密钥分发与认证

密钥分发方法

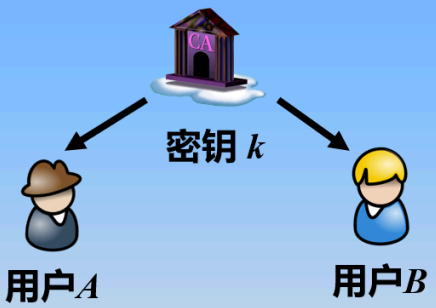
- 假设 A 和 B 有各种密钥分发选择方案

（1）A能够选定密钥并通过物理方法传递给 B



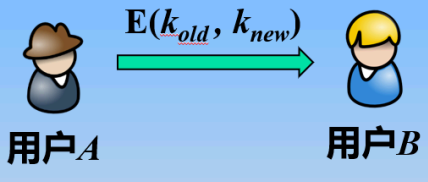
要求手动传递密钥，在链路层加密合理，因为每个链路层加密设备只和此链路另一端交换数据。但是对端到端加密不可行，主机需要不断地跟其它主机和终端交互。

（2）第三方可以选定密钥并通过物理方法传递给A和B



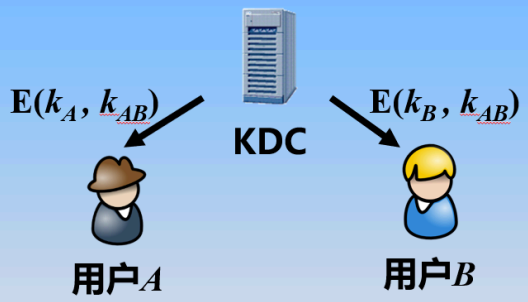
这个方案也要求手动传递密钥，在大范围的分布式系统中很难管理众多密钥的分发。

（3）如果A和B不久之前使用过一个密钥，一方能够把使用旧密钥加密的新密钥传递给另一方。



在这个方案中，如果攻击者成功获得一个密钥，接下来的所有密钥都暴露。

（4）如果A和B各自有一个到达第三方C的加密链路，C能够在加密链路上传递密钥给A和B 。



需要一个可信的密钥分发中心分发会话密钥 k_{AB}。

Kerberos

- 分布式环境面临的威胁
  - 用户可能进入工作站并假装其它用户操作该工作站。
  - 用户可能改变工作站的网络地址并发送伪造的请求。
  - 用户可能监听信息或使用重放攻击，从而获得服务或破坏正常操作。
- 定义：Kerberos是MIT开发的可信密钥服务器系统，在一个分布式网络中提供集中式的私钥第三方认证服务。
- 认证会话：

一个简单的认证会话：

(1) C->AS: $ID_C||P_C||ID_V$

(2) AS->C: Ticket

(3) C->V: ID_C ||Ticket, Ticket=$E(K_V, [ID_C||AD_C||ID_V])$

其中 C = 客户端，AS = 认证服务器，V = 服务器，ID_C = 客户端上用户的身份标识，ID_V = 服务器的身份标识，P_C = 客户端上用户的口令，AD_C = 客户端的网络地址，K_V = 认证服务器和服务器间共享的加密密钥。

  - 存在的问题

- 希望用户需要输入口令的次数最小，然而请求不同服务的时候需要用户输入多次口令。
- 在客户端向AS中请求认证时，口令采用明文传送，容易被窃听。

- 更安全的认证会话

一个更安全的认证会话（引入TGS）：

(1) C->AS: $ID_C||ID_{TGS}$ //每次用户登录会话就执行一次

(2) AS->C: $E(K_C, Ticket_{tgs})$

(3) C->TGS: $ID_C||ID_V||Ticket_{tgs}$ //每个类型的服务执行一次

(4) TGS->C: $Ticket_V$

(5) C->V: $ID_C||Ticket_V$ //每个服务会话执行一次

其中 $Ticket_{tgs}=E(K_{tgs}, [ID_C||AD_C||ID_{tgs}||TS_1||Lifetime_1])$

$Ticket_V=E(K_V, [ID_C||AD_C||ID_V||TS_2||Lifetime_2])$, TGS = 票据授权服务器。

- 存在的问题

这个方案只需要用户输入一次用户口令和保护用户口令。但仍存在以下问题：

- 票据授权票据的有效期问题，容易受到重放攻击。
- 一个网络服务必须确认使用票据的人就是被授予票据的人。
- 服务器可能需要向用户进行自我验证。