iSCSI CHAP 认证

runsisi@hust.edu.cn
http://www.cppblog.com/runsisi

前言

CHAP 认证是 iSCSI 协议中最主要的认证方式，现有的开源 iSCSI 实现（包括 initiator 和 target 实现）基本上都只支持 CHAP 认证，本文将对 CHAP 认证及其在 iSCSI 协议中的应用作一个简单的介绍。

CHAP 协议

CHAP[1]的全称是 Challenge-Handshake Authentication Protocol，协议细节由 RFC 1994 进行定义。CHAP 最初应用在 Point to Point Protocol（PPP）中，用于实现 PPP 服务器对客户端的身份认证，但也可以应用在其它需要对用户或主机进行身份认证的场合，如 iSCSI 协议就规定使用 iSCSI 协议进行通信的设备必须（MUST）实现 CHAP 认证。

对 CHAP 更详细的介绍请参阅维基百科相关的词条[2]及 RFC 1994，下面以路由器 766-1，3640-1 之间的 CHAP 认证交互为例，对 CHAP 认证的交互流程进行简要总结[3]。该例子中 766-1 是被认证方，即需要被认证的客户端，3640-1 是认证方，即处理客户端认证请求的认证服务端。

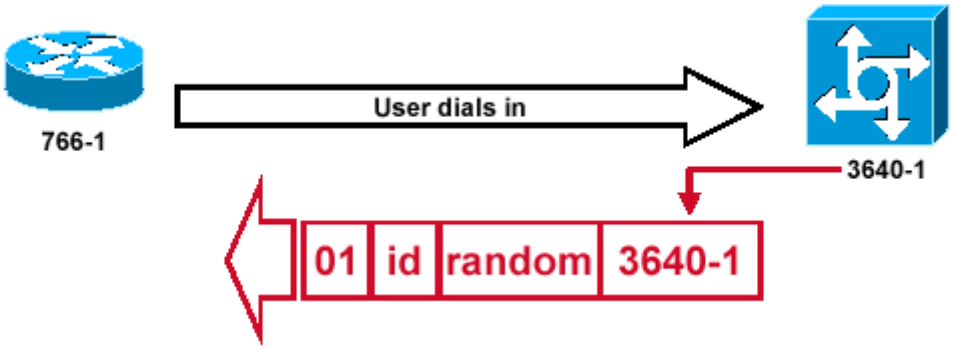

图 1. The Call Comes In

图 1 表示被认证方 766-1 发起对认证方 3640-1 的拨号请求，由于 3640-1 接入该请求的接口配置了必须进行 CHAP 认证，因此 3640-1 将发起 CHAP 挑战（challenge）。

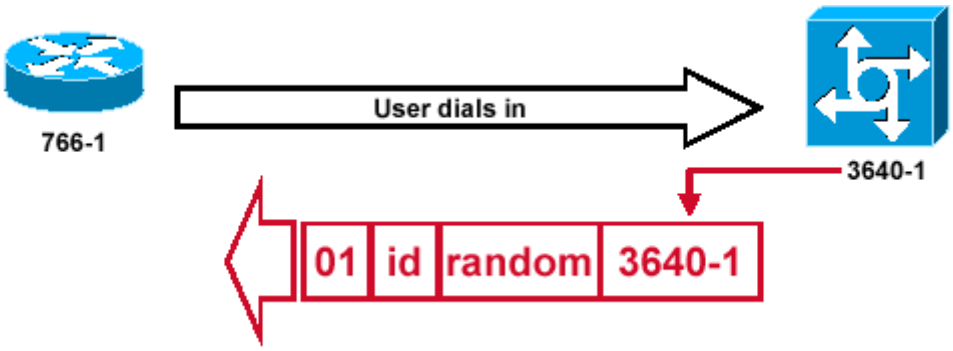

图 2. A Challenge Packet is Built

如图 2，认证方 3640-1 生成一个用于标识该次 CHAP 认证过程的 ID、一串随机数，并与参与认证的用户名一起打包成 CHAP 挑战报文，发送给被认证方。

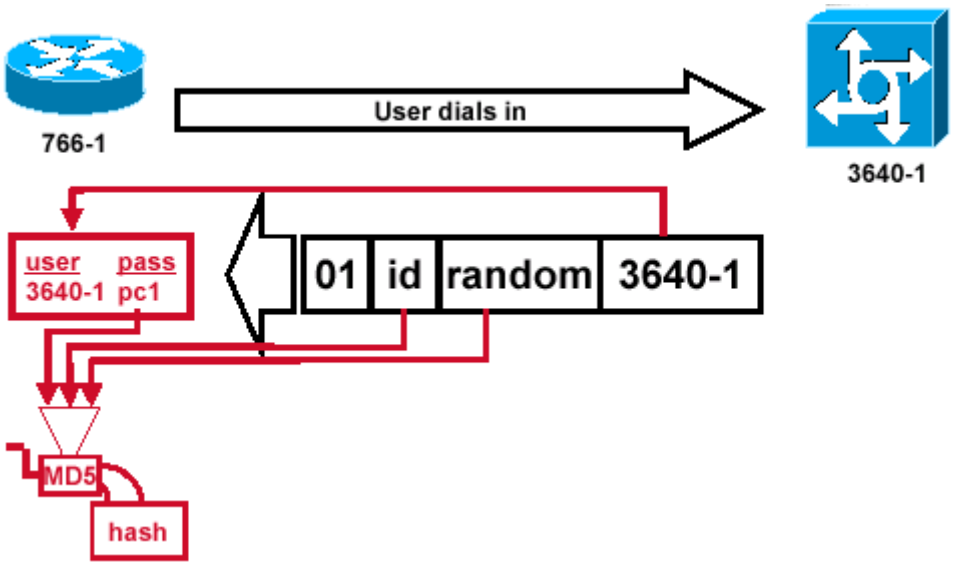

图 3. Receipt and MD5 Processing of the Challenge Packet from the Peer

被认证方 766-1 接收到挑战报文后，对接收到的标识 ID、随机数，以及根据认证方用户名从自己的数据库中查找到的密码用 MD5 算法进行 hash 运算，得到一个 hash 值，该过程如图 3 所示。

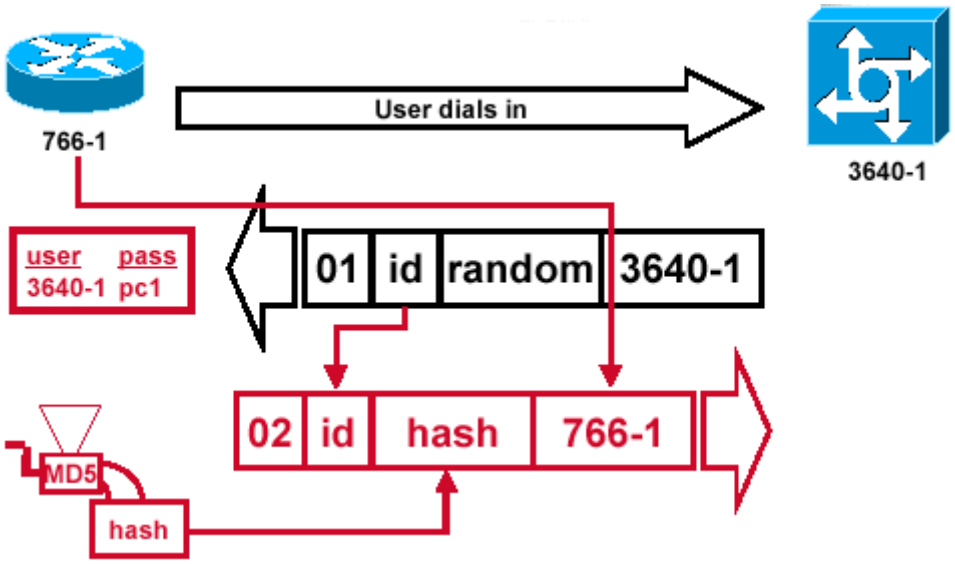

图 4. The CHAP Response Packet Sent to the Authenticator is Built

如图 4，被认证方 766-1 将标识 ID、上一步计算得到的 hash 值以及参与认证的用户名构成一个 CHAP 响应报文发送给认证方。

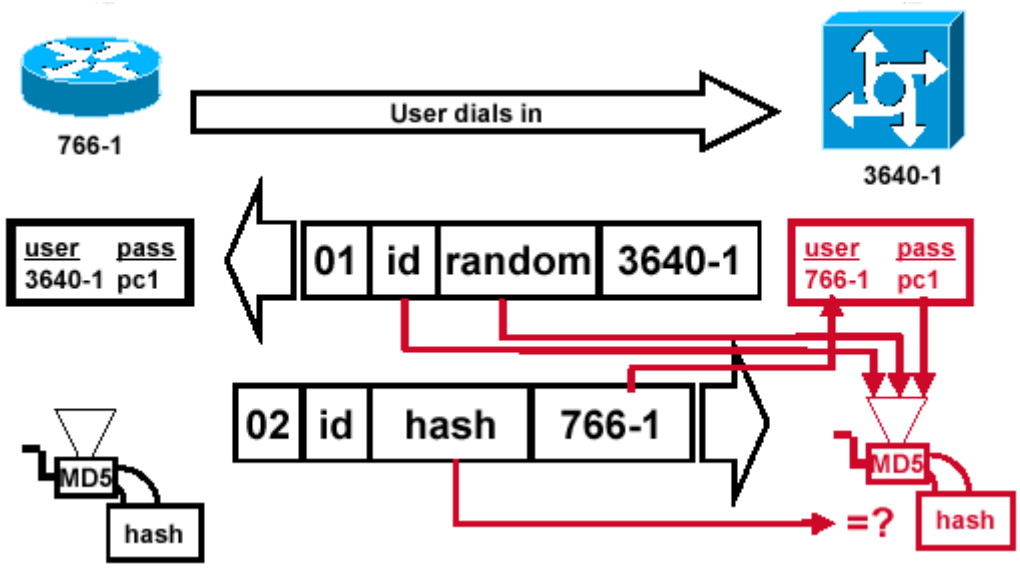

图 5. The Challenger Processes the Response Packet

如图 5，认证方 3640-1 通过响应报文中的标识 ID，找到用来发送挑战时用到的随机树，根据响应报文中被认证方参与认证的用户名从自己的数据库中找到对应的密码，对这三个元素进行同样的 MD5 计算得到一个 hash 值，然后与响应报文中带过的 hash 值进行比较，一致则表明认证通过，并发送认证成功或失败的消息给被认证方，如图 6 所示。

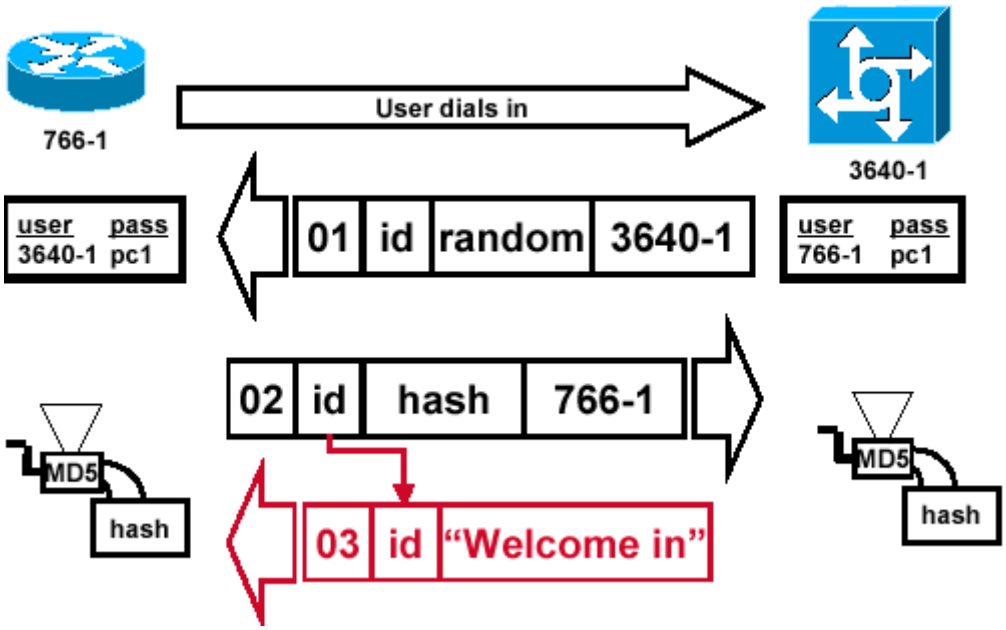

图 6. Success Message is Sent to the Calling Router

CHAP 在 iSCSI 中的应用

RFC 7143 Internet Small Computer Sysem Interface (iSCSI) Protocol[4]对 iSCSI 安全进行了明确的要求，其中 12.1 节提到，“Authentication is OPTIONAL to use but MUST be supported by the target and initiator.”“The initiator and target MUST implement CHAP.”

iSCSI 支持多种形式的认证方式（包括 CHAP，SRP，Kerberos 等，以及各厂商定义的私有认证方式），但除 CHAP 认证之外其它都是可选支持的，主流的开源 iSCSI target 实现，包括 IET[5]，SCST[6]以及 Linux 内核中维护的 Linux-IO[7]都只支持 CHAP 认证。

iSCSI 支持两种形式的 CHAP 认证，1）单向 CHAP 认证，及 2）双向 CHAP 认证。单向认证即仅 target 端对 initiator 端进行认证，双向认证则不仅 target 端要求对 initiator 端进行认证，而且 initiator 端也要求对 target 端进行认证。下面对 iSCSI 中的 CHAP 认证过程进行简单的介绍，更详细的介绍请参考 RFC 7143 12.1.3 节 Challenge Handshake Authenticaion Protocol (CHAP)，以及其附录 B Login Phase Examples。

1）如果在 iSCSI 登录过程中 initiator 和 target 通过协商需要使用 iSCSI 认证，则 initiator 发送：

CHAP_A=<A1, A2…>

其中 A1, A2…为本文第二节提到的 hash 算法的一个可选列表，按优先级进行排序，MD5 算法在 iSCSI 的 CHAP 认证中最为常见。

该过程如图 7 所示。

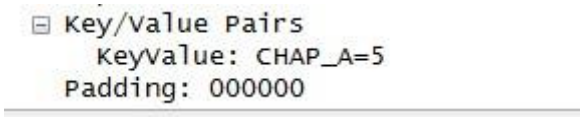

图 7. initiator 发送 hash 算法列表

2）针对该请求，target 要么返回认证失败并拒绝该登录请求，要么发送：

CHAP_A=<A> CHAP_I=<I> CHAP_C=<C>

其中 A 为 initiator 发送的 hash 算法列表中的一个。

该过程如图 8 所示。

图 8. target 发起挑战

3）如果 target 允许登录过程继续进行，则 initiator 发送：

CHAP_N=<N> CHAP_R=<R>

该过程如图 9 所示。

图 9. initiator 发送认证用户名及 hash 值

或者如果 initiator 要求对 target 的身份进行认证，即双向 CHAP 认证的情况，则 initiator 发送：

CHAP_N=<N> CHAP_R=<R> CHAP_I=<I> CHAP_C=<C>

该过程如图 10 所示。

图 10. initiator 发送认证用户名、hash 值及对 target 的认证挑战

4）如果 initiator 身份认证失败，target 必须返回认证失败的状态码并拒绝该登录请求。在双向 CHAP 认证情况下，target 要么返回认证失败的状态码并拒绝该登录请求，要么发送：

CHAP_N=<N> CHAP_R=<R>

该过程如图 11 所示。

图 11. target 发送认证用户名及 hash 值

5）如果 target 身份认证失败，initiator 必须关闭该连接。

上文中提及的 N, (A, A1, A2), I, C 以及 R 分别对应 RFC 1994 中定义的 Name, Algorithm, Identifier, Challenge 以及 Response。

注意在典型的 CHAP 认证过程中，第二步认证方构造挑战报文时需要发送认证方自身的名字，但具体到 iSCSI 应用中，iSCSI 报文并不会构造认证方的名字字段。同样需要注意的是，在典型的 CHAP 认证中，认证方构造挑战报文这一步认证方并不对被认证方做任何形式的校验操作，但在实际的 iSCSI 存储产品中 target 发起 challenge 操作前，一般都会先对 initiator 名字进行校验（initiator 在 login 过程中会告诉 target 自己的名字），如 NetApp 的产品（Clustered Data ONTAP 8.2.1[8]）会校验 initiator 是否配置有相关的 CHAP 认证，如果没有配置则直接返回认证失败，连构造 challenge 报文的步骤都省略了。

参考文献

[1] RFC 1994, http://tools.ietf.org/html/rfc1994

[2] http://en.wikipedia.org/wiki/Challenge-Handshake_Authentication_Protocol

[3] http://www.cisco.com/c/en/us/support/docs/wan/point-to-point-protocol-ppp/25647-understanding-ppp-chap.html

[4] FRC 7143, http://tools.ietf.org/html/rfc7143

[5] http://sourceforge.net/projects/iscsitarget/

[6] http://sourceforge.net/projects/scst/

[7] http://linux-iscsi.org/wiki/LIO

[8] http://www.netapp.com/