Project IV: SOCKS Server (Version 4)

SOCKS 4 Protocol:

http://www.socks.nec.com/protocol/socks4.protocol

SOCKS 4a Protocol (extension)

http://www.socks.nec.com/protocol/socks4a.protocol

In this project, you are asked to implement the SOCKS 4 firewall protocol in the application layer of the OSI model.

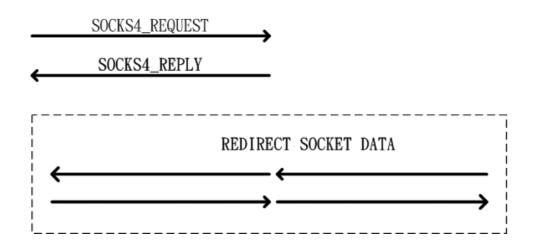
SOCKS is similar to a proxy (i.e. intermediary-program) that acts as both server and client for the purpose of making request on behalf of other clients. Because the SOCKS protocol is independent of application protocols, it can be used for many different services: telnet, ftp, www, etc.

There are two types of the SOCKS operations (i.e.

CONNECT and BIND). You have to implement both of them.

The following figures show the SOCKS operations:

DEST. HOST



## SOCKS4\_REQUEST



VN 4	CD 1 or 2	DST PORT	DST IP = $0.0.0.x$	USER ID	NULL	Domain Name	NULL
1	1	2	4	variable	1	variable	1

#### [CD]

- 1: CONNECT command
- 2: BIND command

## SOCKS4\_REPLY

VN 0	CD 90 or 91	DST PORT	DST IP
1	1	2	4

## [CD]

- 90: request granted
- 91: request rejected or failed

## [DSP IP]

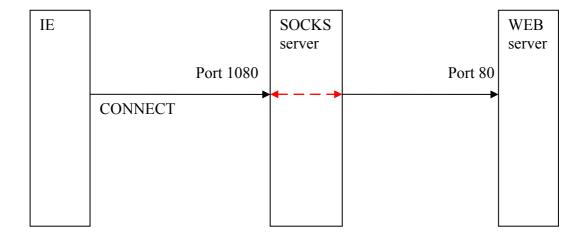
Connect mode: is the DST IP in SOCKS4\_REQUEST

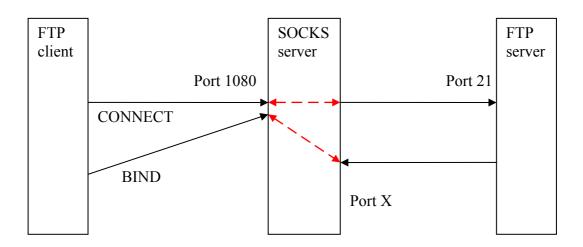
Bind mode: 0

[Port]

Connect mode: is the DST PORT in SOCKS4\_REQUEST

Bind mode: newly binded port in SOCKS server





# SOCKS Version 4 Protocol (CONNECT Operation)

SOCKS 4 SOCKS 4 DEST. CLIENT SERVER HOST ssock = accept(msock) SOCKS4\_REQUEST [\_(CONNECT,\_dst.ip,\_dst.port) \_ user\_id + NULL if (dst. ip == 0.0.0.x)domain\_name + NULL } CHECK FIREWALL RULESET (socks.conf) if (permit\_access) rsock=connectTCP(dst.ip, dst.port) s4\_rep.vn = 0x00; s4\_rep.cd = (rsock > -1) ? 0x5A : 0x5B; s4\_rep.dst\_ipv4 = s4\_req.dst\_ipv4; s4\_rep.dst\_port = s4\_req.dst\_port; SOCKS4 REPLY granted: 0x5A, failed: 0x5BREDIRECT SOCKET DATA WRITE to ssock READ from rsock ssock rsock READ from ssock WRITE to rsock else if (deny\_access) SOCKS4\_REPLY with request rejected:  $\theta x5B$ }

# SOCKS Version 4 Protocol (BIND Operation)

SOCKS 4 SOCKS 4 DEST. CLIENT SERVER HOST ssock = accept(msock) SOCKS4\_REQUEST (BIND, dst.ip, dst.port) user\_id + NULL if (dst. ip == 0.0.0.x)domain\_name + NULL } CHECK FIREWALL RULESET (socks.conf) if (permit\_access) psock=passiveTCP()  $\frac{\text{s4\_rep.vn} = 0x00;}{\text{s4\_rep.cd} = (\text{psock} > -1)?0x5A:0x5B;}$ s4\_rep.dst\_ipv4 = 0; s4\_rep.dst\_port = htons(getsockport(psock)); granted: 0x5A, failed: Ox5B rsock = accept(psock)REDIRECT SOCKET DATA WRITE to ssock READ from rsock ssock rsock READ from ssock WRITE to rsock else if (deny\_access) SOCKS4\_REPLY with request rejected:  $\theta x5B$ }

## Project summary:

- 1. Put the SOCKS client into your CGI program in Project 3 (1). The web page needs extra fields for SOCKS proxy IP and port for each server.
- 2. Write a SOCKS server
- 2.1. CONNECT Operation (use Web Browser)

[socks.conf] : permit c - - - -

2.2. BIND Operation (use FTP client + normal mode)

[socks.conf]: permit b - - - -

3. Include the firewall rule (use Web Browser) in the server.

```
[socks.conf]: permit c - - 140.113. - # [NCTU only] or
```

[socks.conf] : permit c - - 140.114. - # [NTHU only]

## Additional Requirement:

- Use the concurrent, connection-oriented paradigm.
- Output necessary messages on the screen in the SOCKS server.

Due: 2015/1/3 11:59 P.M. (Sunday)