ITBA - "72.07 - Protocolos de Comunicación"

Trabajo Práctico Especial
Definición de protocolo "SCALO-NET"

Alumnos:

De Simone, Franco	61100
Dizenhaus, Manuel	61101
Mattiussi, Agustín Hernán	61361
Sasso, Julián Martín	61535

1 Resumen

El objetivo del protocolo SCALO_NET es facilitar el intercambio de comandos y respuestas a los mismos entre un cliente y un servidor de monitoreo para un proxy SOCKS5. Este protocolo está orientado a texto y funciona sobre TCP.

Una particularidad de SCALO_NET, es que el servidor envía las respuestas en formato CSV para aquellos comandos que solicitan información estadística sobre los usuarios del proxy. El objetivo de esto es facilitarle al cliente el parseo de dichos datos si quisiera visualizarlos en una tabla o planilla de cálculo.

2 Mensajes

En cuanto a las solicitudes y respuestas SCALO_NET, estas siguen (cada una) un formato uniforme, presentado a continuación.

2.1 Comandos

Los comandos son los mensajes que el cliente envía al servidor. Se componen de tres campos:

command	**has_data**	**data**
<cmd-n>	<row-count>	<data>

- **COMMAND:** Número del 0 al 8 en valor ASCII (por ejemplo, '5') que identifica un comando específico. Si en futuras versiones del protocolo se quisieran agregar más comandos (superando tope aparente de 10), bastaría con asignar a los mismos aquellos valores ASCII que suceden a los números. Es decir, ':', ';', pasando por las letras 'a', 'b', etcétera.
- **HAS_DATA:** Este campo indica la cantidad de líneas en "DATA". En la versión actual del protocolo, todos los comandos tienen entre 0 y 1 línea. Sin embargo, por cuestiones de escalabilidad y coherencia con los mensajes de respuesta, se mantiene el significado de este campo.
- **DATA:** Datos específicos del comando. Contiene sólo caracteres ASCII. Su formato específico se desarrolla en cada comando a continuación. Cuando la ejecución del comando no fue exitosa, esta sección contiene el código de error

2.2 Respuestas

Las respuestas son los mensajes que envía el servidor al cliente. Al igual que los comandos, poseen tres campos:

status	**has_data**	**data**
<status-n>	<row-count>	<data>

- **STATUS:** Vale '1' si la ejecución del comando fue exitosa o '0' si no.
- **HAS_DATA:** Número del 0 al 255 (1 octeto) que representa la cantidad de líneas que posee el siguiente campo "DATA". Cada línea se identifica por terminar en '\n'.
- **DATA:** Representa el valor de la respuesta en sí. Contiene sólo caracteres ASCII. Posee tantas líneas como indique "HAS_DATA". Si además la respuesta estuviese en formato CSV, los campos se presentan separados por ';'.

3 Flujo

El flujo del protocolo puede separarse en tres estados:

3.1 Hello

Consiste en un mensaje que envía el servidor al cliente apenas éste establece una conexión. En el campo DATA, indica la versión actual del protocolo.

Formato del mensaje:

status	has_data	data
'1'	1	<version>

3.2 Autenticación

Por una decisión de diseño y seguridad, no cualquier usuario debería poder acceder a los comandos del servidor del protocolo de monitoreo. En una implementación de SCALO_NET, debe definirse en el servidor una contraseña que será compartida por los administradores del proxy SOCKS. La autenticación es el primer comando que el cliente puede enviar, y el único si no logra ingresar la contraseña correcta.

Formato del Mensaje:

command	has_data	data
'0'	1	<password>

3.3 Ejecución

En este estado, tras haberse autenticado, el cliente es libre de intercambiar con el servidor el resto de los comandos. Este último, le enviará la respuesta a su solicitud o un código de error en caso de que la misma hubiese fallado.

Los comandos disponibles, junto con sus valores, son los siguientes:

3.3.1 Añadir Usuario

Permite agregar al proxy SOCKS5 un nuevo usuario con su contraseña. Estos valores se escriben en el campo "DATA", separados por ':'.

status	has_data	data
'1'	1	<user>:<password>

3.3.2 Eliminar Usuario

Permite eliminar un usuario del proxy SOCKS5.

status	has_data	data
'2'	1	<password>

3.3.3 Cambiar contraseña

Permite cambiar la contraseña de un usuario registrado previamente en el proxy SOCKS5. Estos valores se escriben en el campo "DATA", separados por ':'.

status	has_data	data
'3'	1	<user>:<new_password>

3.3.4 Listar las credenciales de POP3

Este comando devuelve una lista en formato CSV con todos los usuarios de POP3 sniffeados por el proxy junto con sus contraseñas.

command	has_data	data
'4'	0	

Una respuesta exitosa a este comando, sería la siguiente:

status	has_data	data
'1'	<row_count>	<csv_data>

En particular, el campo DATA contiene las siguientes columnas, junto con su título, en el siguiente orden:

- Nombre de Usuario
- Contraseña

3.3.5 Obtener métricas

Este comando devuelve una lista en formato CSV con métricas de uso importantes del proxy.

status	has_data	data
'5'	0	<csv_data>

Una respuesta exitosa a este comando sería la siguiente:

status	has_data	data
'1'	<row-count>	<csv-data>

En particular, el campo DATA contiene las siguientes columnas, junto con su título, en el siguiente orden:

- Conexiones actuales al proxy SOCKS5
- Conexiones históricas al proxy SOCKS5

- Conexiones actuales al servidor del protocolo de control
- Conexiones históricas al servidor del protocolo de control
- Conexiones actuales totales
- Conexiones históricas totales
- Cantidad de bytes transferidos por el proxy.

3.3.6 Encender Password Dissectors

Habilita el sniffing de contraseñas de POP3

command	has_data	data
'6'	0	0

3.3.7 Apagar Password Dissectors

Deshabilita el sniffing de contraseñas de POP3

command	has_data	data
'7'	0	0

3.3.8 Listar usuarios del proxy SOCKSv5

Este comando devuelve una lista en formato CSV con los nombres de los usuarios del proxy SOCKS5.

command	has_data	data
'8'	0	

Una respuesta exitosa a este comando sería la siguiente:

status	has_data	data
'1'	<row-count>	<csv-data>

Donde el campo DATA contiene una única columna con los valores de respuesta.

4 Códigos de error

A continuación se explican los códigos de error que podría devolver el servidor ante un fallo en la ejecución de un comando, junto con los valores de los mismos

- ('0') Contraseña Inválida: Contraseña inválida en la autenticación
- ('1') DATA Incompleto: El comando solicitado requiere de algún valor en el campo DATA
- ('2') DATA no vacío: El comando solicitado no debe tener ningún valor en el campo DATA
- ('3') DATA Incorrecto: El formato del campo DATA es incorrecto
- ('4') Usuario Inexistente: El usuario especificado en el comando no existe.
- ('5') Usuario Preexistente: El usuario especificado en el comando ya existía previamente.
- ('6') Límite de Usuarios: Se ha alcanzado el máximo número de usuarios permitido.
- ('7') Error general del servidor.