Laboratorio de Administración y Gestión de Redes y Sistemas
Prueba escrita sobre la teoría. 9 de enero de 2015
Grado en Ingeniería Telemática, Universidad Rey Juan Carlos

Instrucciones:

- Encontrarás en el *home* del puesto del laboratorio el fichero el fichero `~/teoria.tulogin.txt`.

  Cámbiale el nombre, reemplazando *tulogin* por tu verdadero login. Por ejemplo, si eres *jperez*, debes hacer

  ```
  mv teoria.tulogin.txt teoria.jperez.txt
  ```

  Revisa que esté bien hecho. Si te equivocas en este paso tan sencillo, suspenderás.

- Dentro del fichero `teoria.jperez.txt`, escribe tu login, nombre y apellidos y contesta al examen.

Ejercicio 1 (2.5 puntos)

La transparencia 10 del tema 7 dice: *En Unix y Linux se usan mucho otras herramientas de monitorización como Nagios y Zabbix, que ofrecen sus propios servicios pero también incluyen SNMP*

1. Explica este párrafo. ¿por qué se usan otras herramientas? ¿qué significa que ofrezcan sus servicios? ¿qué significa que incluyan SNMP?

2. ¿Va esto contra el estándar?

3. Ventajas e inconvenientes de SNMP frente a herramientas como Nagios y Zabbix. ¿Cuándo usar uno u otro?

Respuesta

1. SNMP es un protocolo que no ha tenido demasiado éxito, tiene muchas limitaciones. Como todo protocolo, pueden usarse desde muchas herramientas, como Nagios, Zabbix y similares. Por ejemplo, podemos tener una impresora que exporte su estado mediante SNMP, y monitorizarlo desde Zabbix.

   Pero estas herramientas no solo soportan el protocolo estándar de SNMP, sino que también ofrecen sus propios agentes, que usan su propio protocolo, más potente y flexible.

   Si no hablamos de un host sencillo como una impresora, sino de otro más complejo, como un ordenador, también podremos exportar su estado mediante un agente (cliente) SNMP. Pero esto no es muy habitual, posiblemente preferiremos emplear el cliente específico de nuestra herramienta de monitorización.

2. Si solo usamos agentes SNMP, estamos siguiendo el estándar. Pero si se usan agentes particulares (cosa muy frecuente), sí, esto va contra el estándar.

3. SNMP es universal, vamos a encontrar un agente SNMP en cualquier dispositivo con un mínimo de calidad y capaz de conectarse a Internet. Otras herramientas más avanzadas usan sus propios protocolos no universales. Típicamente se usa SNMP cuando no hay otra opción disponible, por ejemplo impresoras, switches, cámaras IP, dispositivos domóticos, etc. En estos dispositivos no vamos a encontrar soporte nativo para, por ejemplo, Zabbix. Cuando se trate de monitorizar ordenadores convencionales, sí podremos usar agentes propios de nuestra herramienta.

Ejercicio 2 (2.5 puntos)

Un servidor de DHCP puede estar configurado en modo *authoritative* o en modo *non authoritative*

1. ¿Qué significa esto?

2. ¿Quién decide que un servidor esté en un modo u otro?

3. ¿Qué garantías tiene un cliente de que un servidor *authoritative* lo es realmente? ¿Cómo evitar que un servidor *non authoritative* conteste como *authoritative*?

4. ¿Qué problemas puede causar un servidor de DHCP que tenga este parámetro mal configurado?

Observación: La transparencia 8 del tema 6 explica el comportamiento de un servidor *authoritative* y *non authoritative* ante un DHCPREQUEST. Pero **no** es eso lo que debes contestar. Cíñete al enunciado del ejercicio.

Respuesta

1. Una respuesta con el flag *authoritative* activado significa que el servidor declara que su respuesta es *oficial*, el servidor está indicando que él es quien tiene autoridad en ese segmento de red.

   Una respuesta *non authoritative*, el servidor no asegura ser la autoridad oficial.

   Cuando un cliente hace una petición *DISCOVERY*, si hay varios servidores recibirá varias respuestas *OFFER*, la norma permite que el cliente elija la respuesta más adecuada, según su criterio. (Debería preferir una con autoridad frente a una sin autoridad, aunque muchas implementaciones se quedan con la primera en llegar)

   Cuando un cliente pide una dirección y al servidor le parece incorrecta, si tiene autoridad se *atreve* a contradecirle. Si el servidor no tiene autoridad, no responde.

2. El administrador de la máquina donde está el servidor de DHCP. Debería ser el administrador de la red, pero no tiene por qué serlo. Es imposible evitar que un servidor se atribuya la autoridad que no tiene.

3. Un servidor de DHCP mal configurado puede ser muy problemático. Supongamos un atacante que instala un servidor en su portátil y lo lleva a una red ajena. El servidor puede empezar a repartir direcciones incorrectas, puede inutilizar la red con parámetros incorrectos, o peor aún, *infectar* la red proporcionando direcciones de servidores con malware.

   Si las repuestas del atacante son *non authoritative* y el servidor de la red víctima también envía respuestas *non authoritative*, los clientes podrían usar indistintamente una u otro. Al igual que el caso en que atacante y víctima declaran ser *authoritative*.

   El peor caso es el de una red víctima donde el servidor tenga la configuración por omisión (*non authoritative*), y el atacante esté dando respuestas (*authoritative*). Las víctimas preferirán al atacante.

   El caso menos dañino es el de un atacante, posiblemente involuntario, que tenga un servidor por omisión (*non authoritative*). Si el administrador de la red ha tenido la precaución de configurar el servidor oficial como (*authoritative*), los clientes preferirán el oficial.

   Esto es lo previsto en el estándar, pero como hemos dicho, muchas implementaciones atienden la primera respuesta recibida, sea cual sea, por lo que aceptar la respuesta correcta o la del atacante acaba resultando aleatorio.

Ejercicio 3 (2.5 puntos)

Probablemente conoces la *flamenca del whatsapp*, un *emoji* (icono) que representa una mujer vestida de rojo, bailando.

¿Sería posible escribir en Linux un texto que la incluya? ¿Qué haría falta?

Respuesta

Unicode es un estándar de codificación de caracteres. Es muy completo, incluye no solo todos los caracteres de prácticamente cualquier lenguaje humano, sino también iconos. Por ejemplo, desde la versión 6.0 (año 2010), incluye la *flamenca*, que es el carácter `'DANCER' (U+1F483)`

Toda herramienta que soporte unicode nos permitirá manejar este y cualquier otro carácter, aunque para poder visualizarlo, será necesario tener instalado el *font* adecuado. Los fonts pueden cambiar de aspecto, por ejemplo este icono en algunos fonts representa un hombre de color amarillo.

Ejercicio 4 (2.5 puntos)

Un script de shell bash normalmente empieza por `#!/bin/bash`

1. ¿Por qué?
2. ¿Qué sucede en caso contrario?

3. ¿Qué sucede con el intento de ejecución de un script sin permiso de ejecución?

Respuesta

1. Para indicar explícitamente donde está el ejecutable que sabe interpretar ese texto. La secuencia almohadilla-admiración es un *número mágico*, un convenio que indica que si estos son los primeros caracteres de un fichero, lo que aparece a continuación es el path del intérprete.

2. Que el sistema no sabría cuál es el intérprete, intentaría ejecutarlo con el intérprete por omisión que haya especificado el usuario. Si ambos coinciden, todo funciona bien. En otro caso, para ejecutarlo sería necesario invocar al intérprete (bash) y luego pasarle el script, bien por la entrada estándar, bien como primer argumento.

3. Depende. Si se intenta ejecuta el script directamente, no funcionará. Pero si se invoca a la shell y se le pasa el script, se ejecutará normalmente.