Seguridad en Sistemas Informáticos (SSI)

Seguridad centrada en el nodo

Carlos Pérez Conde

Departament d'Informàtica

Escola Técnica Superior d'Enginyeria

Universitat de València

Guión

- **Seguridad física**
- Autentificación y control de acceso
- Implicaciones de seguridad de los servicios
- Control de acceso mediante envolventes (*wrappers*)
- Auditoría y registros
- Integridad del sistema
- Detección de intrusos

Seguridad física (I)

- **Seguridad física = barreras que protegen la consola**
  - no debe ser el punto más débil de la cadena de seguridad
- **Mantener un plan de seguridad física**
- **Protección de los componentes físicos**
  - Mantener un entorno adecuado (monitorización continua)
    fuego, humo, polvo, terremotos, explosiones, temperaturas extremas, insectos, ruido eléctrico, rayos, vibraciones, humedad, agua...
  - Prevenir accidentes
    comida y bebida, golpes casuales, caídas de equipos, enganchones con cables...
  - Cuidado con el acceso físico
    suelos elevados y falsos techos, conducciones de aire, paredes transparentes...
  - Vandalismo
  - Prevenir robos y limitar los daños que pueden causar
    anclar los sistemas, encriptación, ojo con portátiles y PDAs, equipos de repuesto...

Seguridad física (II)

- **Protección de los datos**
  - Escuchas a escondidas (eavesdropping)
    directamente, de la red, de las emisiones electromagnéticas, puertos auxiliares en terminales
  - Protección de las copias de seguridad
    verificación, protección física, destruir el medio antes de desecharlo, encriptación
  - Protección del almacenamiento local
    impresoras y buffers de impresión, particiones de intercambio
  - Terminales desatendidas
    auto-desconexión, salva-pantallas con contraseña

Guión

- Seguridad física
- **Autentificación y control de acceso**
- Implicaciones de seguridad de los servicios
- Control de acceso mediante envolventes (*wrappers*)
- Auditoría y registros
- Integridad del sistema
- Detección de intrusos

Gestión de cuentas

- **Altas**
  - No activar cuentas sin contraseña (o con contraseña trivial)
  - Mejor utilizar grupos que utilizar cuentas compartidas
- **Bajas**
  - Desactivar inmediatamente las cuentas que dejen de usarse
  - Cambiar todas las contraseñas que el que se va sabía (idealmente: cambiar todas las cuentas)
- **Evitar las cuentas compartidas (mejor usar grupos)**
  - difuminan la responsabilidad
  - difíciles de controlar

Riesgos de las contraseñas

- **Fáciles de adivinar por personas**
  - Triviales: nombre de usuario, password, passwd...
  - Datos personales que otras personas pueden conocer/averiguar: amigos, familiares, personajes, libros, películas favoritas...
  - Contraseñas por defecto, ejemplos (de este curso, de un libro...)
- **Fáciles de adivinar por ordenadores**
  - Cualquier palabra que pueda aparecer en un diccionario
  - Cualquier modificación de éstas: reordenación, sustitución de caracteres, desplazamientos en orden alfabético
  - Contraseñas cortas y/o con caracteres poco variados
    Ej: 4 minúsculas -> 26^4 (menos de 0'5 millones de posibilidades)
- Observables
  - Mirando por encima del hombro, cámaras...
  - Observando el tráfico de red (seleccionable)

Buenas contraseñas

- **Difíciles de adivinar**
  - Contienen mayúsculas, minúsculas, números y/o caracteres de puntuación
  - Contienen más de 7 caracteres
- **Fáciles de recordar**
  - Se evita tener que escribirlas
- **Fáciles de teclear**
  - Evitar miradas por encima del hombro
  - Fácil detectar pruebas en la cuenta

Sugerencias y ejemplos

- **Juntar 2 ó más palabras con caracteres especiales**
  - Se-me=olvidará?
- **Iniciales de una frase fácil de recordar**
  - EcEidR.¿Os? (Esta contraseña es imposible de recordar. ¿O sí?)
- **Fórmulas pseudo-matemáticas:**
  - 2+uno*cinco=7
- **Combinar varios métodos**
  - Combino: 2+tres=5 ¿Sí?

Contraseñas

- **Con adivinar una sola contraseña es suficiente:**
  - Cuentas sin contraseña y cuentas joe (login = password)
  - Fuerza bruta: ataques de diccionario
    - puede hacerse fuera de línea (basta con una copia de /etc/passwd)
    - se prueban:
      - palabras de diccionarios y enciclopedias
      - transformaciones: permutaciones, añadir dígitos... (2400 reglas es habitual)
    - programas: **Crack**, de Alec Muffett; **John the Ripper**, de Solar Designer (http://www.openwall.com/john/)
    - John the Ripper sobre sobre las contraseñas de prácticas:
      - en 49 s: 5 adivinados
      - en 3 min, 29 s: 17
- **Contramedidas**
  - Usar buenas contraseñas: educar a los usuarios, usar crackers
  - Ficheros de contraseñas *shadow*
  - Otros métodos: contraseñas de un solo uso, tarjetas, criptografía...

Guión

- Seguridad física
- Autentificación y control de acceso
- **Implicaciones de seguridad de los servicios**
- Control de acceso mediante envolventes (*wrappers*)
- Auditoría y registros
- Integridad del sistema
- Detección de intrusos

Servicios comunes

- **Servicios más comunes**
  - systat, ident, finger, DNS, auth, NTP, SNMP, RIP
  - telnet, RPC, rexec, rlogin, rsh, X
  - SMTP, POP, IMAP, NNTP, TFTP, FTP, rcp, UUCP, HTTP
  - MUDs, IRCs...

Implicaciones de seguridad: Casos de ejemplo

- **systat (salida de: *who, w, ps...*), prueba: telnet host 11**
  - proporciona información valiosa para los atacantes
- **telnet: conexión con autentificación mediante contraseña**
  - packet sniffing → contraseñas de un solo uso, criptografía
    - en la red local
    - en cualquier punto intermedio de la conexión (p.e: en un ISP)
  - secuestro de sesiones → criptografía
- **Network Time Protocol (NTP)**
  - cambio no autorizado de la hora del sistema:
    → ataques de repetición (ej: Kerberos, tickets válidos durante un tiempo)
    → fecha errónea en los registros del sistema
    → órdenes programadas con cron ó at pueden no ejecutarse (ej: detección de intrusos, copias de seguridad...)

Implicaciones de seguridad: Casos de ejemplo: httpd

- **Componentes básicos:**
  - servidor, CGIs
  - clientes, java, aplicaciones locales
- **Desafíos a la seguridad:**
  - se pueden explotar las vulnerabilidades del servidor y de los CGI: DoS, acceso no autorizado al sistema, obtención del control del servidor
  - distribución no autorizada de información almacenada en el servidor (ej: números de tarjetas de crédito)
  - interceptación de la información intercambiada entre clientes y servidores
  - el servidor puede explotar vulnerabilidades en los clientes: DoS, obtención de información, acceso al sistema...
  - desafíos introducidos por software complementario: bases de datos, lectores de documentos...

Más información

- Recomendado: cap. 17, "TCP/IP Services" de "Practical Unix..."
- Recomendación: cap. 18, "WWW security", de "Practical Unix..." y fuentes adicionales descritas en la pág. 539.

Caso de ejemplo: SSH

- **SSH: conexiones remotas y otros servicios seguros sobre una red insegura. (draft-ietf-secsh-architecture-07.txt)**
- **Componentes:**
  - Protocolo de la capa de transporte [SSH-TRANS]
    - autentificación del servidor (criptografía asimétrica)
    - confidencialidad (criptografía simétrica)
    - integridad
    - compresión (opcional)
  - Protocolo de autentificación del usuario [SSH-USERAUTH]
    - permite al servidor autentificar al usuario del cliente
    - métodos: criptografía asimétrica, contraseña, basada en nodo
  - Protocolo de conexión (SSH-CONN)
    - multiplexa varios canales lógicos en el túnel encriptado:
      Intérpretes interactivos, reenvío de puertos TCP/IP arbitrarios y conexiones X11.

Autentificación del servidor

- **cada nodo debe tener una clave por cada algoritmo de clave pública requerido (actualmente DSS [FIPS-186])**
- **el cliente recibe esta clave pública para que autentifique al servidor**
- **comprobación de la correspondencia entre nombre de host y clave pública**
  - comparación con una base de datos local
  - certificado por una Agencia de Certificación
  - opcionalmente: no comprobarlo en la primera conexión
    (peligro: posible ataque de hombre en el medio)

Una sesión con SSH

- **Establecimiento de la conexión: sobre TCP/IP**
  - intercambio de la versión de protocolo (1 ó 2)
  - intercambio de capacidades, acuerdo de algoritmos:
    clave del servidor, encriptación, integridad y compresión
  - intercambio de claves:
    - autentificación del servidor por parte del cliente
    - acuerdo de un secreto K y un extracto (*hash*) H

DATOS SIN ENCRIPTAR

    - a partir de K y H se generan las claves criptográficas para los alg. acordados
    - H es el identificador de sesión
  - autentificación del cliente
  - creación de canales y sesiones:
    interactivas, ejecución remota de órdenes, reenvío de X, reenvío de conexiones TCP
  - ejecución de un intérprete de órdenes (o de una orden)

DATOS ENCRIPTADOS

- **Finalización de la conexión**
  - termina el programa del cliente (intérprete u orden), y
  - terminan todas las conexiones adicionales (ej: conexiones X11 redirigidas)

Caso de ejemplo: OpenSSH

- **http://www.openssh.org (parte de OpenBSD)**
- **Clientes**
  - ssh: cliente que reemplaza a rlogin y telnet
  - scp: cliente que reemplaza a rcp
  - sftp: cliente que reemplaza a ftp
- **Servidores**
  - sshd: servidor para todos los clientes
  - sftp-server: servidor auxiliar para sftp
- **Gestión de claves para autentificación**
  - ssh-keygen: generación, administración y conversión de claves
  - ssh-add: añade identidades RSA o DSA para el agente de autentificación
  - ssh-agent: agente de autentificación

Utilización de ssh (I)

Conexión interactiva

```
$ id
uid=1000(usu1) gid=1000(usuarios) groups=1000(usuarios)
$ ssh pruebas.uv.es
      Conectarse a pruebas.uv.es como usu1
$ ssh root@pruebas.uv.es
      Conectarse a pruebas.uv.es como root
```

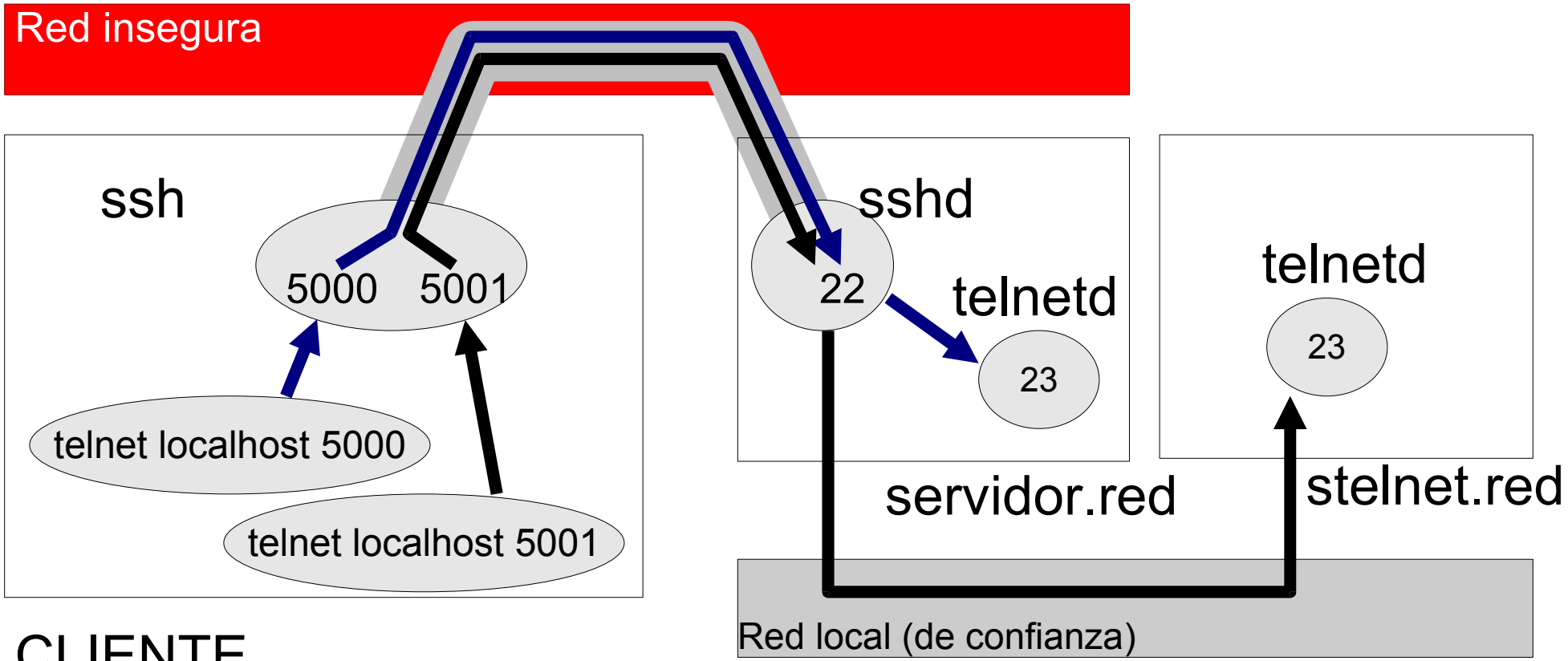
Ejecución remota de órdenes

```
$ ssh usu2@pruebas.uv.es w
12:48pm  up 6 days,  4:11,  3 users,  load average: 0.03, 0.04, 0.00
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU  WHAT
carlos   pts/0    -                11:09am  1:39m  0.01s  0.01s  /bin/cat
carlos   pts/2    -                12:44pm  0.00s  0.11s  0.03s  w
$ ssh -f pruebas.uv.es xclock
$ (cd / && tar cf - etc) | ssh pruebas tar xf -# HOME
$ tar cf - www | ssh pruebas (cd /tmp && tar xf - )  # /tmp
bash: syntax error near unexpected token ‘(c’
$ tar cf - www | ssh pruebas "(cd /tmp && tar xf - )"# /tmp
$ tar cf - www | ssh pruebas "\"(cd /tmp && tar xf - )\""  # /tmp
bash: (cd tmp && tar xf - ): command not found
```

Utilización de ssh (II)

Redirección de puertos locales

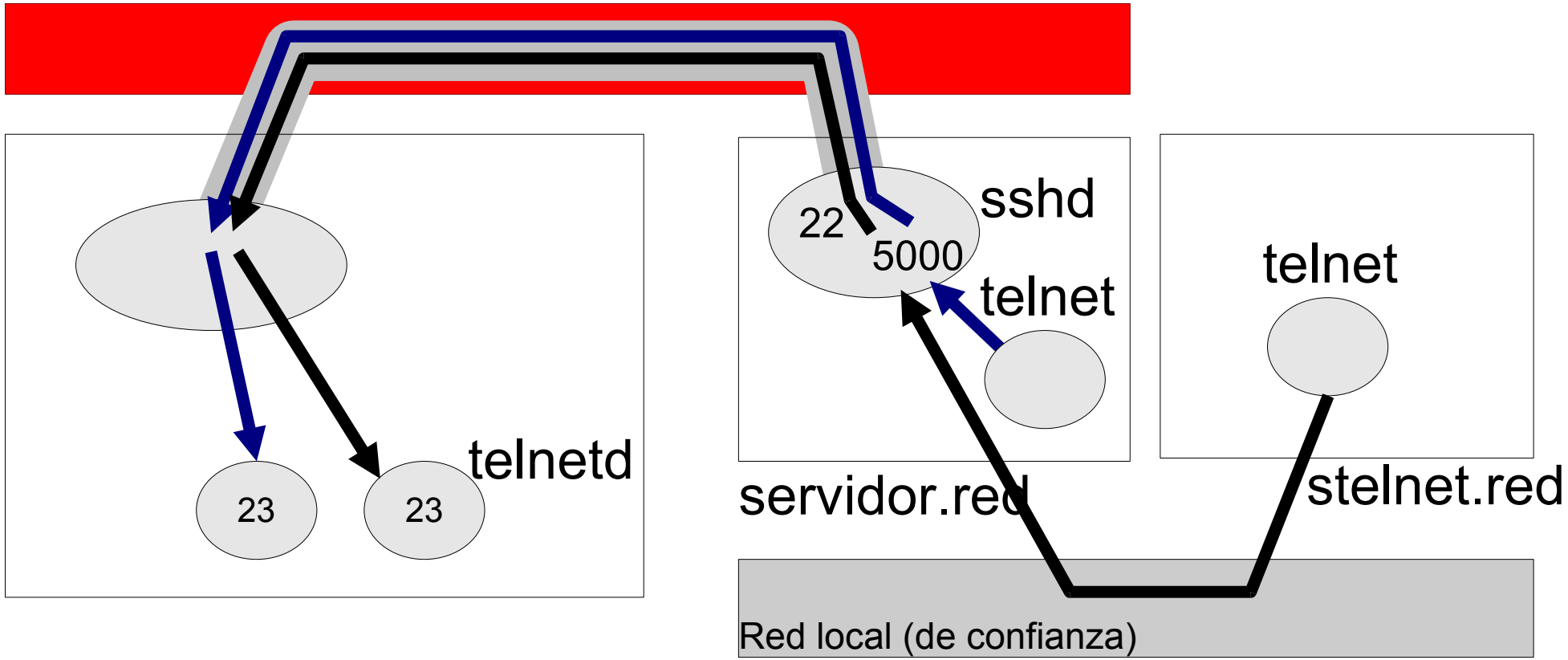
```
$ ssh -f -N -L5000:localhost:23 -L5001:stelnet.red:23 servidor.red
```



Utilización de ssh (III)

Redirección de puertos remotos

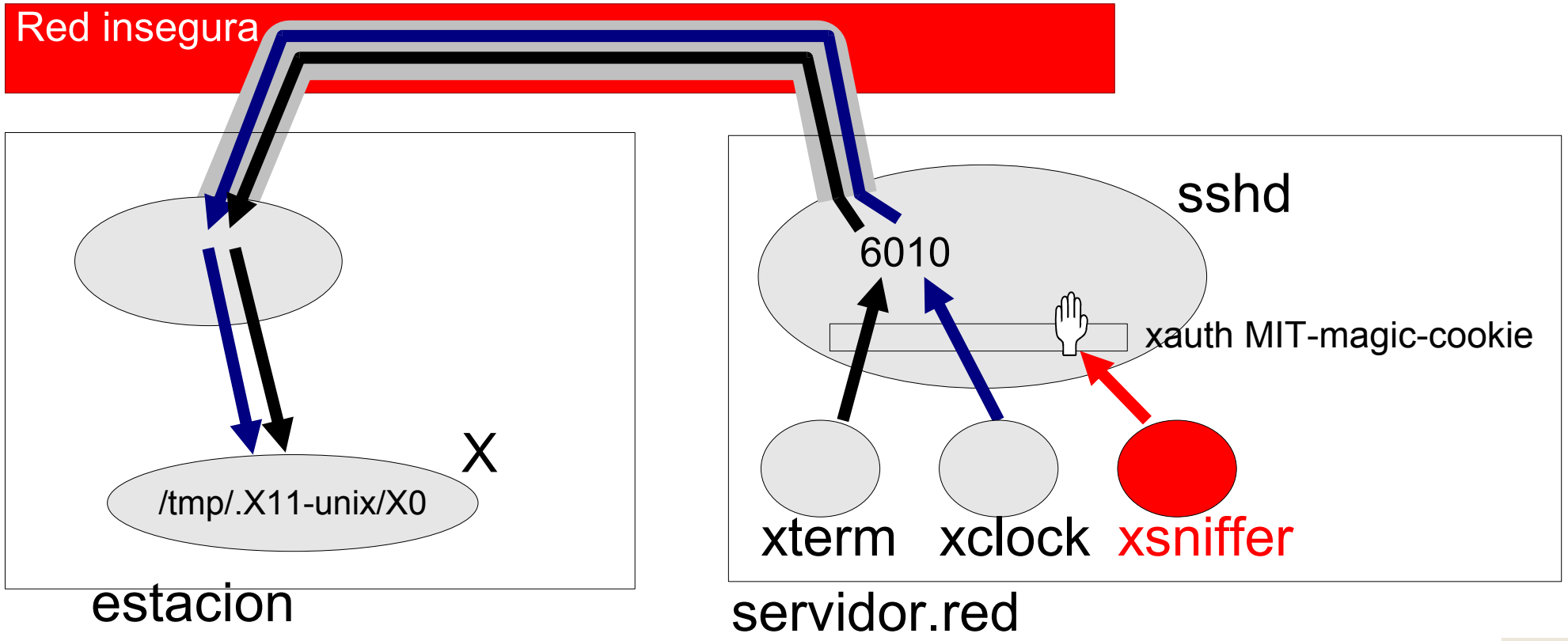
```
$ ssh -f -N -R5000:localhost:23 servidor.red
```



Utilización de ssh (IV)

Redirección de X

```
estacion_$ ssh servidor.red
servidor_$ echo $DISPLAY
servidor.red:10.0
servidor_$ xclock & xterm &
```



Configuración de sshd (v.2)
Opciones en /etc/ssh/sshd_config

- AllowUsers, AllowGroups, DenyUsers, DenyGroups
- ChallengeResponseAuthentication (skey), HostbasedAuthentication, KerberosAuthentication, PasswordAuthentication, PubkeyAuthentication
- PermitRootLogin, PermitEmptyPasswords
- Banner
- Ciphers (aes128-cbc,3des-cbc,blowfish-cbc,cast128-cbc,arcfour)
- MACs (hmac-md5,hmac-sha1,hmac-ripemd160...)
- AllowTCPForwarding, X11Forwarding

Configuración de sshd (v.2)
Fichero AUTHORIZED_KEYS

- $HOME/.ssh/authorized_keys
- [opciones] tipo de clave (ssh-dss o ssh-rsa) clave codificada (base64) comentario
- opciones: from, command, environment, no-port-forwarding, no-X11-forwarding...

Ejemplo condensado

```
ssh-dss AAAAB3N...KN carlos@maquina

command=”dump /home”,no-pty,no-port-forwarding ssh-dss XX...2= Backup
```

Configuración de sshd (v.2)
Otros ficheros

- **Claves**
  - /etc/ssh/ssh_host_key, /etc/ssh/ssh_host_dsa_key, /etc/ssh/ssh_host_rsa_key
  - /etc/ssh/ssh_host_key.pub, /etc/ssh/ssh_host_dsa_key.pub, /etc/ssh/ssh_host_rsa_key.pub
- **Computadores conocidos**
  - /etc/ssh/ssh_known_hosts and $HOME/.ssh/known_hosts
- **Otros**
  - /etc/nologin
  - /etc/hosts.allow, /etc/hosts.deny
  - $HOME/.ssh/environment
  - $HOME/.ssh/rc, /etc/ssh/sshrc

Configuración de ssh (v.2)
Opciones: /etc/ssh/ssh_config y ~/.ssh/config

- **Selección de opciones (man ssh)**
  - Host - restricción de aplicabilidad - patrones con * y ?
  - Ciphers (aes128-cbc...), HostKeyAlgorithms (ssh-rsa, ssh-dss)
  - Compression, CompressionLevel
  - ForwardX11, GatewayPorts, LocalForward, RemoteForward
  - MACs (hmac-md5,hmac-sha1,hmac-ripemd160, ...)
  - StrictHostKeyChecking, PreferredAuthentications, PubkeyAuthentication...

Ejemplo

```
Host prueba prueba.uv.es
IdentityFile2 ~/.ssh/clave_usu_openssh_dsa
IdentityFile2 ~/.ssh/clave_root_openssh_dsa

Host *
ForwardX11 no
PubkeyAuthentication no
```

Guión

- Seguridad física
- Autentificación y control de acceso
- Implicaciones de seguridad de los servicios
- **Control de acceso mediante envolventes (wrappers)**
- Auditoría y registros
- Integridad del sistema
- Detección de intrusos

Control de acceso: Encapsuladores (wrappers)

- **¿Qué son? Programas que permiten controlar el acceso a otros programas**
  - en caso de acceso permitido: ejecutar un programa
  - en cualquier caso: registrar el hecho y sus circunstancias
- **¿Por qué utilizar encapsuladores?**
  - reunión en un solo programa de toda la lógica de seguridad → mayor facilidad de comprobación
  - actualización independiente de los programas encapsulados
  - control de la información que llega a los servidores
- **Ejemplos:**
  - smap/smapd: para sendmail, por TIS (Trusted Information Systems)
  - tcpwrapper: propósito general, para UDP y TCP, por Wietse Venema
  - xinetd:

tcpwrappers (I)

- **Permite:**
  - mostrar un mensaje (banner) al cliente
  - realizar una búsqueda inversa doble de la dirección IP del cliente, cortando la conexión en caso de discrepancia
  - control de acceso en base al nodo cliente y el servicio solicitado
  - emplear ident (RFC 1413) para averiguar el nombre del usuario
  - registrar información mediante syslog
  - opcionalmente, ejecutar órdenes
  - transferir el control al verdadero servidor de red
  - transferir el control a un entorno "trampa"
- **Configuración: /etc/hosts.allow y /etc/hosts.deny**
  - Si hosts.allow la permite explícitamente, se permite la conexión
  - Si no, si hosts.deny la deniega explícitamente, se deniega
  - Se permite la conexión
- **Instalación:**
  - instalar tcpd y modificar inetd.conf
  - trasladar los demonios, instalar tcpd como los demonios

tcpwrappers (II)

- **Entrada en hosts.{allow|deny}**
  daemon_list: client_host_list : option : option : ...
  - daemon_list: lista de servidores (argv[0]) o comodines
  - client_host_list: lista de nombres o direcciones de nodo, patrones o comodines
  - patrones
    - para nombres: .uv.es incluye slabii.uv.es, bugs.informat.uv.es, etc.
    - para direcciones: 147.156. incluye a 147.156.16.1, 147.156.17.95
    - para redes: 147.156.16.0/255.255.254.0 incluye a las redes 147.156.16.0 y 146.156.17.0
  - comodines (operadores: EXCEPT)
    - ALL: siempre encaja
    - LOCAL: nombres sin punto
    - KNOWN/UNKNOWN: usuarios cuyo nombre se conoce/desconoce
    - PARANOID: nodos cuyo nombre no encaja con su dirección
  - expansiones posibles:
    - %a (%A): dirección del cliente (servidor)
    - %c: información del cliente (usuario@nodo)

tcpwrappers (III)

- **Opciones (cont.)**
  - de control de acceso: allow, deny
  - de ejecución de otras órdenes
    - spawn shell_comand - spawn (/alt/safe_finger -l @%h | /usr/ucb/mail root) &
    - twist shell_command - twist /bin/echo 421 Mensaje de error
    - de red: keepalive, linger
  - de identificación de usuario: rfc931 [timeout_in_seconds]
  - otras:
    - banners /some/directory - volcar el contenido de daemon al cliente
    - nice [number] - cambiar el valor nice del servidor
    - setenv name value
    - umask 022
    - user nobody
    - user nobody.kmem

tcpwrappers (IV)

Ejemplo de configuración avanzada: hosts.allow

```
all : .informat.uv.es : allow
in.telnetd, in.ftpd : seg.fiable.uv.es : allow
in.fingerd : all : spawn (/alt/safe_finger -l@%h | /bin/mail root) & :
deny
all : all : banners /root/tcpd/banners : deny
```

Herramientas auxiliares: tcpdchk y tcpdmatch

```
# tcpdchk
warning: /etc/hosts.allow, line 28: ypserv: no such process name
in /etc/inetd.conf

# tcpdmatch in.telnetd carlos@slabii.informat.uv.es
client:     hostname    slabii.informat.uv.es
client:     address     147.156.17.60
client:     username    carlos
server:     process     in.telnetd
matched:    /etc/hosts.deny line 11
access:     denied
```

Control de acceso: xinetd (I)

- **xinetd - extended internet services daemon**
  - http://www.xinetd.org/
  - http://www.linuxfocus.org/English/November2000/article175.shtml
- **¿Qué ofrece xinetd?**
  - inetd + tcpd
  - control de acceso basado en franjas horarias
  - registro completo de conexiones (tanto de éxitos como de fracasos)
  - contención frente a ataques DoS (de denegación de servicio)
    - limitar el nº de servidores simultáneos de cada tipo
    - limitar el nº total de servidores
    - limitar el tamaño de los ficheros de registro
  - asociación de servicios a interfaces específicas de red
  - puede ser usado como proxy

xinetd (II)

Configuración

```
defaults / service nombre_de_servicio
{
        atributo operador{=, +=, -=} valor(es)
        ...
}
```

- atributos:
  - registro de información:
    - log_type: SYSLOG selector [level] ó FILE [max_sz [abs_max_sz]]
    - log_on_success: PID, HOST, USERID (RFC1413), EXIT, DURATION
    - log_on_failure: HOST, USERID, ATTEMPT, RECORD
  - control de acceso:
    - no_access, only_from list_of_clients : 147.156.17.1, 147.156.17.0, 147.156.{16,17}, 147.156.16.0/23, red, slabii.uv.es, .uv.es
    - port, protocol, socket_type, type
    - interface, access_times
    - banner, banner_success, banner_fail

xinetd (III)

- **atributos (cont.)**
  - control de la ejecución:
    - server, server_args
    - nice, env, passenv, user, group
    - wait
    - redirect
  - contención del uso de recursos:
    - cps límite secs - limitar el nº de conexiones por segundo
    - instances n - limitar el nº de servidores simultáneos
    - max_load n - limitar la carga máxima que debe soportar un servidor (SO)
    - per_source n - limitar el nº de conexiones desde un mismo origen al servidor

xinetd (IV)

Ejemplo de configuración

```
defaults
{
      instances = 60
      log_type = SYSLOG
authpriv
      log_on_success = HOST PID
      log_on_failure = HOST
RECORD
      only_from = localhost
}
```

```
service ftp
{
      socket_type = stream
      wait = no
      user = root
      server = /usr/sbin/in.ftpd
      server_args = -l -a
      log_on_success += DURATION
USERID
      log_on_failure += USERID
      nice = 10
}
```

Ejemplo de registro

```
# cat /var/log/secure
Jan 30 13:06:00 slabii xinetd[409]: FAIL: ftp address from=147.156.17.60
Jan 30 13:06:00 slabii xinetd[21712]: USERID: ftp OTHER :carlos
Jan 30 13:06:07 slabii xinetd[409]: START: ftp pid=21715 from=127.0.0.1
Jan 30 13:06:07 slabii xinetd[21715]: USERID: ftp OTHER :carlos
Jan 30 13:07:30 slabii xinetd[409]: EXIT: ftp pid=21715 duration=83(sec)
```

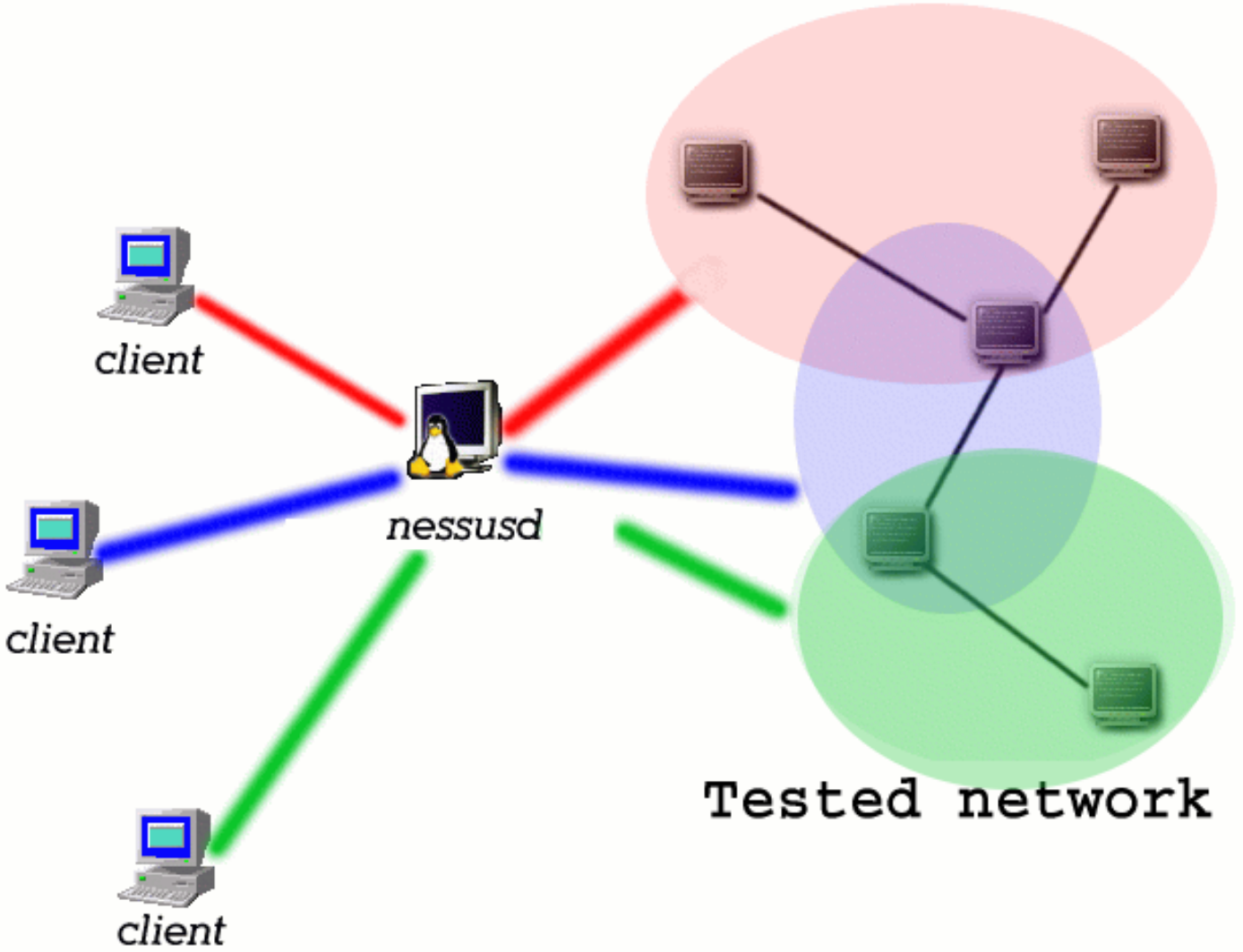
Guión

- Seguridad física
- Autentificación y control de acceso
- Implicaciones de seguridad de los servicios
- Control de acceso mediante envolventes (*wrappers*)
- **Auditoría y registros**
- Integridad del sistema
- Detección de intrusos

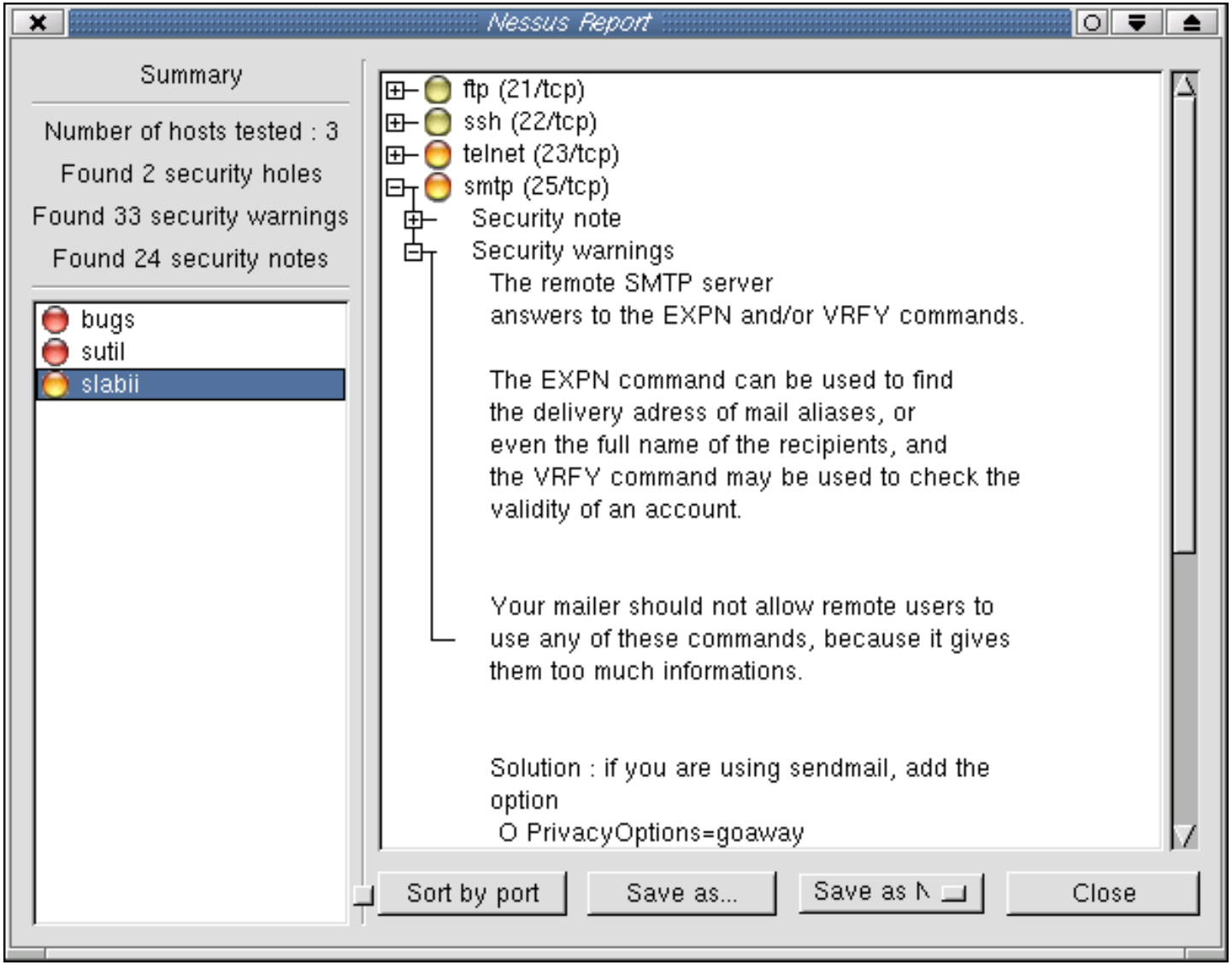
Herramientas de auditoría
Ej: NESSUS

- arquitectura modular y abierta: plug-in externos, cliente/servidor
- NASL: Nessus Attack Scripting Language
- base de datos de seguridad actualizada diariamente
- exploración concurrente de múltiples nodos
- exploración con credenciales
- informes completos (con soluciones) y exportables (ASCII, LaTeX, HTML...)
- gratuito, con soporte comercial

NESSUS: arquitectura



NESSUS: ejemplo de informe



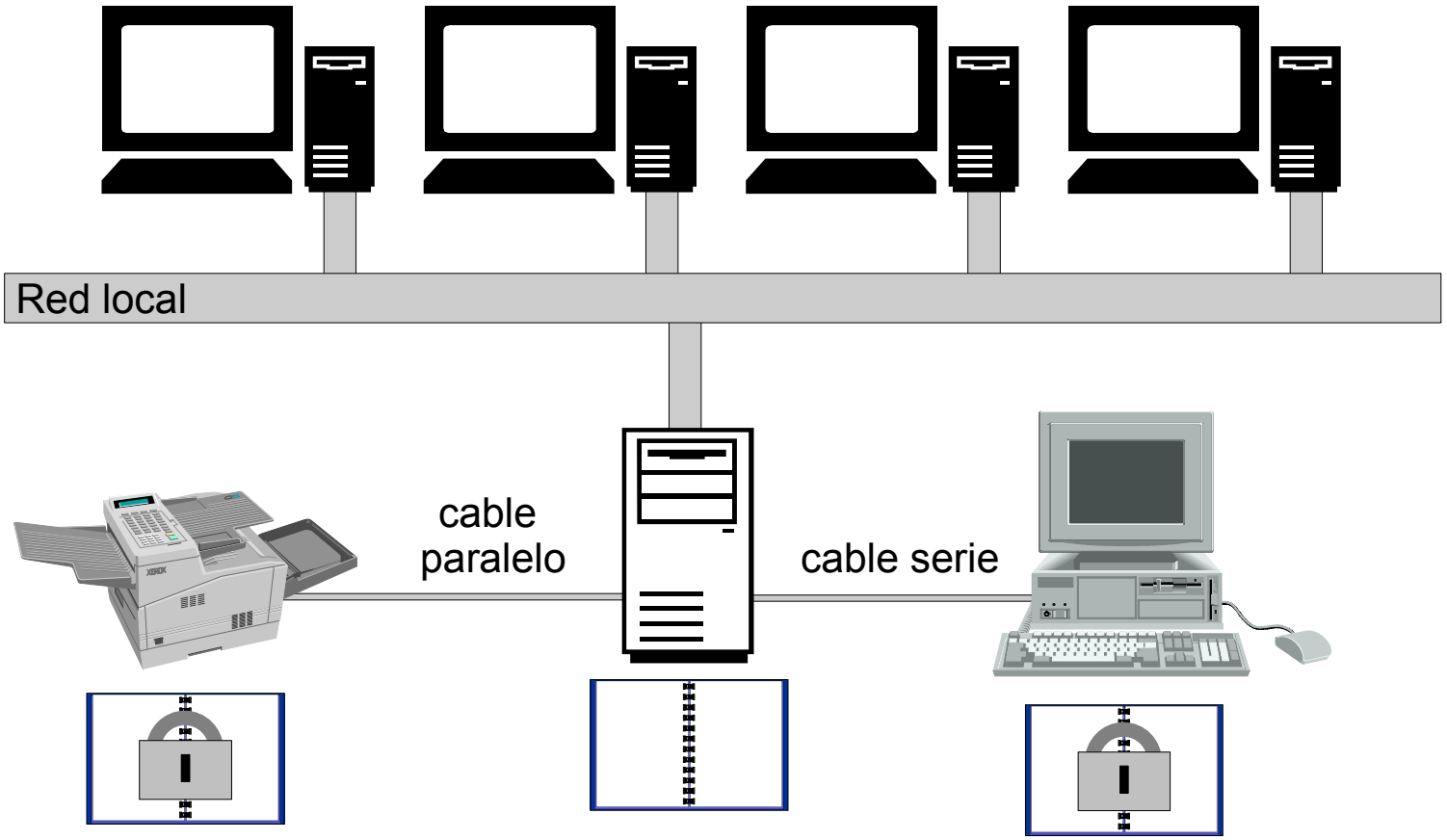
NESSUS: conclusiones

- **Las herramientas de auditoría de seguridad**
  - estimación parcial de la seguridad de la red
    (ej: los CGIs locales no pueden ser auditados)
  - responde a: ¿es segura mi red en este momento?
    (cada día aparecen nuevas vulnerabilidades que pueden cambiar la respuesta)
  - necesitan ejecución frecuente
- **Resumiendo. Las auditorías de seguridad:**
  - pueden poner de manifiesto problemas
  - que no detecten ninguno NO IMPLICA que no pueda haberlos
  - NO PUEDEN ser la única medida de seguridad
  - SON ÚTILES como parte de una política de seguridad completa

Registros del sistema

- **Contenido: guardan la historia del sistema**
  - Conexiones y desconexiones: quién, desde dónde, a qué hora...
  - Ejecución de programas
  - Conexiones a otros sistemas...
  - Intentos fallidos de todo lo anterior
- **Utilidad**
  - Detectar y corregir problemas de configuración y/o funcionamiento
  - Detectar y reaccionar ante una violación de la política de seguridad
    - Estimar el alcance de la agresión: reconstruir el sistema
    - Obtener ayuda del servicio técnico
    - Llevar a cabo una investigación, servir como prueba, cobrar una póliza de seguros
    - Demandar al agresor
- **Debilidad: pueden ser alterados o modificados**
  - Soluciones: copias remotas, copias impresas...

Almacenamiento seguro de registros



Registros en Unix (I)

- **Ficheros almacenados en: /usr/adm, /var/adm, /var/log**
- **Ficheros más habituales**
  - utmp, utmpx, wtmp, wtmpx — conexiones y desconexiones
  - lastlog — última conexión de cada usuario
  - sulog — cambios de identidad mediante su
  - messages, SYSLOG ... — mensajes del sistema
  - acct, pacct — contabilidad de procesos
- **Otros ficheros más específicos**
  - maillog — mail
  - ftplog, xferlog — ftp
  - htmlaccesslog — HTTP
  - ...

Registros en Unix (II)
Conexiones (I)

Usuarios conectados actualmente: who, w, finger

```
# who -THl
USER     MESG LINE     LOGIN-TIME   FROM
psorian   -   pts/12   Jan  8 16:44 (213.0.68.67)
jmaestr   -   pts/2    Jan  8 17:15 (usuario1-36-187-54.dialup.uni2.es)
psorian   -   pts/14   Jan  8 17:16 (213.0.68.67)
jmallac   +   pts/4    Jan  8 17:33 (213.96.69.115)

# w
  5:34pm  up 23 days, 22:41, 10 users,  load average: 2.63, 2.36, 2.21
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU  WHAT
psorian  pts/12   213.0.68.67       4:44pm 44.00s  1.00s  0.88s  sqlplus
jmaestr  pts/2    usuario1-36-187-  5:15pm  0.00s  0.13s  0.06s  joe avl.h
psorian  pts/14   213.0.68.67       5:16pm  0.00s  0.19s  0.15s  joe
     ind...
jmallac  pts/4    213.96.69.115     5:33pm  2.00s  0.12s  0.12s  -sh

# finger carlos
Login: carlos                           Name: Carlos Pérez Conde
Directory: /home/carlos                 Shell: /bin/bash
On since Tue Jan  9 08:23 (CET) on :0 (messages off)
On since Tue Jan  9 08:24 (CET) on pts/0   6 hours 21 minutes idle
No mail.
No Plan.
```

Registros en Unix (III)
Conexiones (II)

Última conexión de cada usuario: lastlog

```
# lastlog -u carlos
Username         Port     From             Latest
carlos           :0                        mar ene  9 08:23:57 +0100 2001
```

Consulta del registro de conexiones: last

```
   # last psorian | head -3
psorian  pts/14  213.0.68.67      Mon Jan  8 17:16   still logged in
psorian  pts/12  213.0.68.67      Mon Jan  8 16:44   still logged in
psorian  pts/11  ruth.irobot.uv.e Mon Jan  8 14:16 - 14:18  (00:02)
```

Estadísticas de tiempo de uso del sistema: ac

```
# ac -pd | egrep "psorian|lalario|jorts|total"
        psorian                                  0.02
        lalario                                  4.80
Jan  7  total       27.41
        psorian                                 11.78
        lalario                                 12.93
        jorts                                    0.82
Today   total       59.74
```

Registros en Unix (IV)
Actividad de los usuarios

Registro del intérprete de órdenes: ~/.sh_history ~/.bash_history

```
# tail -2 ~carlos/.bash_history
ftp ftp.funet.fi
gpg --verify linux-2.4.0.tar.gz2.sign linux-2.4.0.tar.bz2
```

Registro de ejecución de programas: lastcomm

```
# lastcomm mail -f /var/log/acct
mail                   egalleg  ??         0.00 secs Tue Jan  9 12:22
mail                   jgomar   ??         0.01 secs Tue Jan  9 11:25
mail             S     root     ??         0.00 secs Tue Jan  9 04:02
```

Generación de resúmenes: sa, sar

```
# sa acct -m | head -2
root    3610 2123743.09re     472.03cp         0avio       431k
mcid     156     622.79re       0.94cp         0avio       937k
# sar -u 10 2
Linux 2.2.16-22 (slabii.informat.uv.es)         01/09/01
16:22:21          CPU     %user     %nice   %system     %idle
16:22:31          all     99.70      0.00      0.30      0.00
16:22:41          all     99.40      0.00      0.60      0.00
Average:          all     99.55      0.00      0.45      0.00
```

Registros en Unix (V)
syslog (I)

- **syslog: redirecciona la información que recibe**
  - recibe de: el núcleo de Unix, demonios del sistema, aplicaciones...
  - envía a: ficheros, dispositivos, usuarios, syslogs remotos...
- **Mensajes para syslog:**
  - nombre del programa, facilidad, prioridad, texto del mensaje
  - `Jan  9 16:16:18 slabii ftpd[23876]: FTP session closed`
  - facilidades: kern, user, auth, lpr... uucp, local0... local7
  - prioridades: emerg, alert, crit... info, debug
- **Ficheros:**
  - /etc/syslog.conf configuración
  - /dev/log socket Unix, mensajes de procesos locales
  - /dev/klog ídem. para mensajes del núcleo
  - puerto UDP 514 mensajes remotos

Registros en Unix (VI)
syslog (II)

Ejemplo de configuración: /etc/syslog.conf

```
# Enviar los mensajes del núcleo a la consola
kern.*      /dev/console

# Enviar todo lo de nivel info o superior, salvo el mail y los mensajes
# de autentificación privados, al fichero messages
*.info;mail.none;authpriv.none      /var/log/messages

# Enviar los mensajes de autentificación privados a un fichero de
# acceso más restringido
authpriv.*  /var/log/secure

# Guardar todo lo del mail en maillog
mail.*      /var/log/maillog

# Enviar una copia de todo a nuestro gestor central de registros
*.*   @regs,@regs-backup

# Imprimir una copia de los mensajes de autorización
auth,authpriv.*   /dev/lp0
```

Registros en Unix (VII)
Gestión de registros

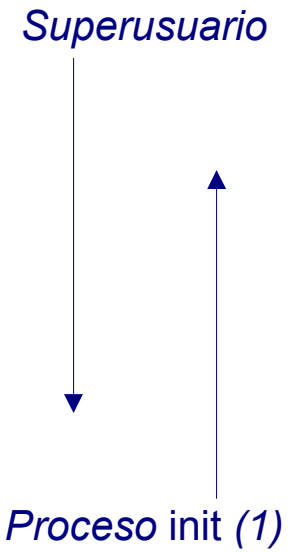
- **Justificar (y limitar) la confianza que se deposita**
  - los registros pueden ser alterados y/o borrados
  - que esté en el registro no implica que haya sucedido
  - que no esté en el registro no implica que no haya sucedido
- **Limitar el tamaño de los registros**
  - realizar rotaciones periódicas (ej: logrotate)
  - realizar copias de seguridad suficientemente frecuentes
    - para mantener un registro completo
    - para protegerse frente al borrado y/o la modificación accidental
- **Revisar frecuentemente su contenido**
  - filtrar eliminando específicamente las entradas no interesantes
  - revisar periódicamente versiones completas de los registros
    - herramientas automáticas
    - guiones/programas personalizados

Guión

- Seguridad física
- Autentificación y control de acceso
- Implicaciones de seguridad de los servicios
- Control de acceso mediante envolventes (*wrappers*)
- Auditoría y registros
- **Integridad del sistema**
- Detección de intrusos

Ficheros inmutables y de sólo añadir

- **Implementados en BSD 4.4 (FreeBSD, NetBSD, BSDI)**
- **Adecuados para:**
  - Inmutables: ficheros de configuración (/etc/rc), dispositivos...
  - Sólo añadir: registros del sistema
- **Implementación:**



Niv. Seg.	*Modo*	*Significado*
-1	Permanentemente inseguro	Unix estándar
0	Inseguro	Se pueden cambiar los modos de inmutabilidad y de sólo añadir
1	Seguro	No se pueden cambiar estos modos Los dispositivos montados, /dev/mem y /dev/kmem son de sólo lectura.
2	Altamente seguro	Como el seguro; pero con todos los ficheros especiales de dispositivos en modo de sólo lectura.

Sistemas de ficheros de sólo lectura

- **Nadie puede modificar un CD-ROM usando un lector**
  - Otros: discos duros con protección contra escritura por *hardware*, algunos discos extraíbles (p. ej.: las unidades ZIP)
- **Dividir los ficheros del sistema**
  - modificables
  - no modificables
- **Ventajas**
  - Menor necesidad de copias de seguridad
  - Mayor facilidad de administración
  - Cuotas innecesarias
  - No es necesario limpiar periódicamente estos sistemas de ficheros

Sistemas de ficheros de sólo lectura

- **Inconvenientes/limitaciones**
  - No es adecuado para la protección de los datos de los usuarios (suelen cambiar con demasiada frecuencia)
  - La mayor parte de los discos duros no permiten esta opción
  - Todo el disco debe ser protegido (espacio desaprovechado)
  - Son necesarios al menos dos discos duros por máquina
  - Los lectores de CD-ROM son más lentos que los discos duros

Detección de cambios
Comparación de copias

- **Ventajas**
  - Es el método más directo y seguro
  - Permite la restauración del original en caso de cambio
- **Inconvenientes**
  - Es necesario el doble de espacio
  - La comparación byte a byte es costosa
  - No detecta: cambios en los derechos de acceso, o de dueño
- **Ojo:**
  - Usar programas fiables para la comparación
  - Asegurarse de que la comparación se lleva a cabo
  - Es necesario asegurarse de que la copia no puede ser modificada
    - Copias locales: usar discos extraíbles o mantenerlas encriptadas
    - Copias remotas: en sistemas más fiables que el que se desea verificar

Detección de cambios
Resúmenes y Metadatos

- **Listado simple**
  - *ls -ild*
    incluye: número de nodo-i, permisos, nº de alias, dueño, grupo, tamaño, fecha de la última modificación y nombre
- **Estructura de directorios**
  - Los directorios importan. P. ej.: /etc con 777 permite:
    - Sustituir /etc/passwd con otro con una cuenta de root sin contraseña
    - Convertirse en root
    - Reemplazar el original
  - Una comparación de ficheros (incluso binaria) no lo detectaría
- **Resúmenes**
  - Es fácil modificar ficheros sin alterar sus atributos
  - Es fácil modificar ficheros conservando el CRC de *sum*
  - Es necesario usar resúmenes menos vulnerables: MD5

Guión

- Seguridad física
- Autentificación y control de acceso
- Implicaciones de seguridad de los servicios
- Control de acceso mediante envolventes (*wrappers*)
- Auditoría y registros
- Integridad del sistema
- **Detección de intrusos**

AIDE

http://www.cs.tut.fi/~rammer/aide.html

- **AIDE (Advanced Intrusion Detection Environment):**
  - Instalación: generación de resúmenes de los ficheros del sistema y almacenamiento en una base de datos propia
  - Uso habitual: generación de resúmenes, comparación con los de referencia, notificación de cambios
- **Permite/Ofrece**
  - Emplear diferentes algoritmos para resúmenes (*CRC32, MD5, SHA1, tiger*)
  - Portabilidad, funciona sobre Solaris, Linux, FreeBSD, OpenBSD...
  - Configurabilidad
  - Código fuente disponible
- **A tener en cuenta/Limitaciones**
  - Debe ser utilizado de forma conjunta con otras medidas de seguridad
  - Depende de la integridad de la base de datos

AIDE: políticas (fichero aide.conf)

- **Entradas:**
  - `[/|!|=] regexp [expresión] [# comentario]`
    `ej: /bin p+i+n+u+g+s+m+md5`
  - expresión: [grupo[{+|-}grupo]...]
  - grupos predefinidos

p	bits de control de acceso
i	número de nodo-i
n	número de alias (hard links)
u	usuario
g	grupo
s	tamaño
m	mtime
...	
S	comprobar que el tamaño aumenta
md5	resumen MD5
...	

```
# cat aide.conf
/ R
!/etc/ntp.drift
!/home
!/var
=/tmp
#
```

...

R	[R]ead-only: p+i+n+u+g+s+m+c+md5
L	[L]og file: p+i+n+u+g
E	[E]mpty group
>	monotonically growing file: p+u+g+i+n+S

AIDE: ejemplo de informe

```
# aide –-check
AIDE found differences between database and filesystem!!
Start timestamp: 2003-11-21 13:12:24
Summary:
Total number of files=4837,added files=0,removed files=0,changed files=3

Changed files:
changed:/tmp/src
changed:/tmp/src/linux-2.4.20-xfs/include/linux
changed:/tmp/src/linux-2.4.20-xfs/include/linux/random.h
Detailed information about changes:

Directory: /tmp/src
  Mtime    : 2003-03-30 14:21:00                , 2003-11-21 12:25:34
  Ctime    : 2003-11-21 12:22:57                , 2003-11-21 12:25:34
Directory: /tmp/src/linux-2.4.20-xfs/include/linux
  Mtime    : 2003-03-26 16:00:12                , 2003-11-21 13:12:12
  Ctime    : 2003-11-21 12:22:57                , 2003-11-21 13:12:12
File: /tmp/src/linux-2.4.20-xfs/include/linux/random.h
  Mtime    : 2000-01-25 23:13:46                , 2003-11-21 13:12:12
  Ctime    : 2003-11-21 12:22:56                , 2003-11-21 13:12:12
  Inode    : 25316                              , 25748
  MD5      : qX+nZdDWEzFkAxKY6K6/pg==           , YxQCZ6YBbZpbRzHZ7alIOg==
```

OSSEC HIDS

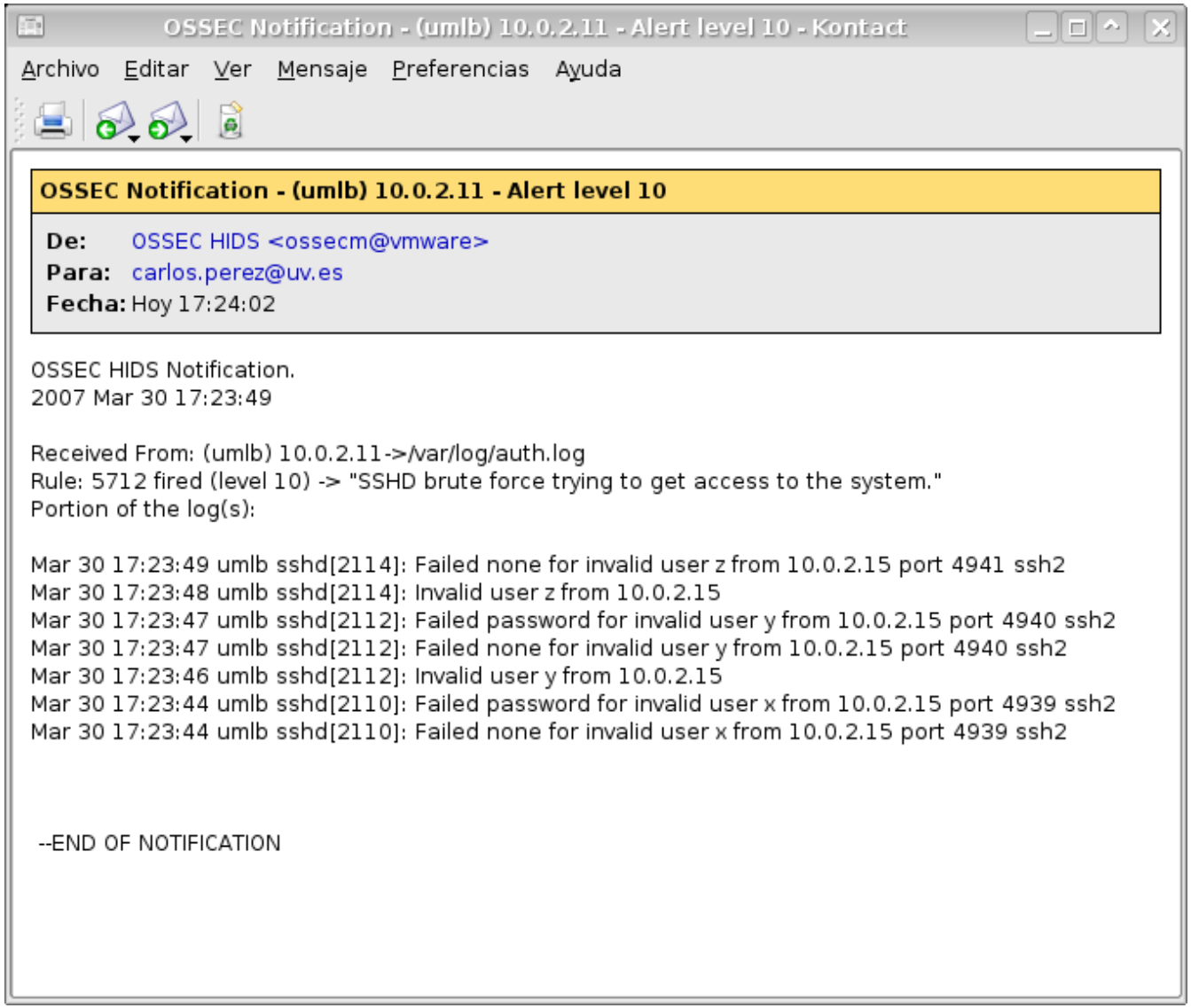
http://www.ossec.net

- **Análisis de registros**
  - aplicaciones Unix (pam, su, sudo, adduser, logins...)
  - servidores (SSH, samba, ftp, mail, web)
  - cortafuegos (iptables, windows...), NIDS (snort), utilidades (nmap...)
  - Windows (registro, registro de eventos, encaminamiento...)
- **Comprobación de integridad (syscheck, similar a Aide)**
- **Detección de rootkits (rootcheck)**
- **Alertas en base a secuencias temporales**
- **Respuesta activa: *tcpwrappers*, filtrado de paquetes**
- **Disponible para varias plataformas**
- **Opciones: local, servidor/agente**

OSSEC
ejemplo de detección y respuesta

```
umlf:~# hostname
umlf

umlf:~# ssh x@umlb
x@umlb's password:
Permission denied, please try again.
x@umlb's password:
Permission denied, please try again.
x@umlb's password:
Permission denied (publickey,password).

umlf:~# ssh y@umlb
(igual que antes)

umlf:~# ssh z@umlb
(idem)

umlf:~# ssh a@umlb
(no responde)
```

OSSEC
ejemplo de detección y respuesta



OSSEC
ejemplo de detección y respuesta

```
umlb:~# tail -1 /etc/hosts.deny
ALL:10.0.2.15

umlb:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       0    --  umlf.uml.ssi         anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
DROP       0    --  umlf.uml.ssi         anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

umlb:~#
```