FACULTAD DE INGENIERIA U.N.A.M.
DIVISION DE EDUCACION CONTINUA

CENTRO DE INFORMACION Y DOCUMENTACION
"ING. BRUNO MASCANZONI"

El Centro de Información y Documentación Ing. Bruno Mascanzoni tiene por objetivo satisfacer las necesidades de actualización y proporcionar una adecuada información que permita a los ingenieros, profesores y alumnos estar al tanto del estado actual del conocimiento sobre temas específicos, enfatizando las investigaciones de vanguardia de los campos de la ingeniería, tanto nacionales como extranjeras.

Es por ello que se pone a disposición de los asistentes a los cursos de la DECFI, así como del público en general los siguientes servicios:

- **Préstamo interno.**
- **Préstamo externo.**
- **Préstamo interbibliotecario.**
- **Servicio de fotocopiado.**
- **Consulta a los bancos de datos: librunam, seriunam en cd-rom.**

Los materiales a disposición son:

- **Libros.**
- **Tesis de posgrado.**
- **Noticias técnicas.**
- **Publicaciones periódicas.**
- **Publicaciones de la Academia Mexicana de Ingeniería.**
- **Notas de los cursos que se han impartido de 1980 a la fecha.**

En las áreas de ingeniería industrial, civil, electrónica, ciencias de la tierra, computación y, mecánica y eléctrica.

El CID se encuentra ubicado en el mezzanine del Palacio de Minería, lado oriente.

El horario de servicio es de 10:00 a 19:30 horas de lunes a viernes.

FACULTAD DE INGENIERIA U.N.A.M.
DIVISION DE EDUCACION CONTINUA

A LOS ASISTENTES A LOS CURSOS

Las autoridades de la Facultad de Ingeniería, por conducto del jefe de la División de Educación Continua, otorgan una constancia de asistencia a quienes cumplan con los requisitos establecidos para cada curso.

El control de asistencia se llevará a cabo a través de la persona que le entregó las notas. Las inasistencias serán computadas por las autoridades de la División, con el fin de entregarle constancia solamente a los alumnos que tengan un mínimo de 80% de asistencias.

Pedimos a los asistentes recoger su constancia el día de la clausura. Estas se retendrán por el periodo de un año, pasado este tiempo la DECFI no se hará responsable de este documento.

Se recomienda a los asistentes participar activamente con sus ideas y experiencias, pues los cursos que ofrece la División están planeados para que los profesores expongan una tesis, pero sobre todo, para que coordinen las opiniones de todos los interesados, constituyendo verdaderos seminarios.

Es muy importante que todos los asistentes llenen y entreguen su hoja de inscripción al inicio del curso, información que servirá para integrar un directorio de asistentes, que se entregará oportunamente.

Con el objeto de mejorar los servicios que la División de Educación Continua ofrece, al final del curso deberán entregar la evaluación a través de un cuestionario diseñado para emitir juicios anónimos.

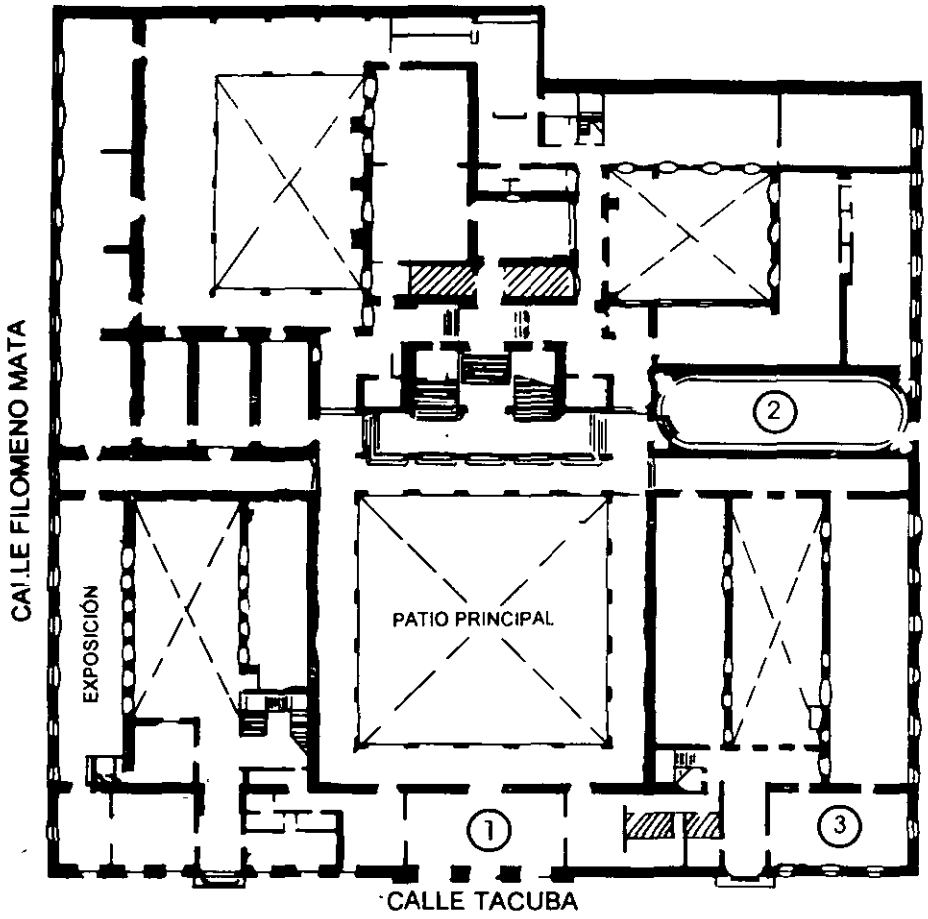
Se recomienda llenar dicha evaluación conforme los profesores impartan sus clases, a efecto de no llenar en la última sesión las evaluaciones y con esto sean más fehacientes sus apreciaciones.

Atentamente

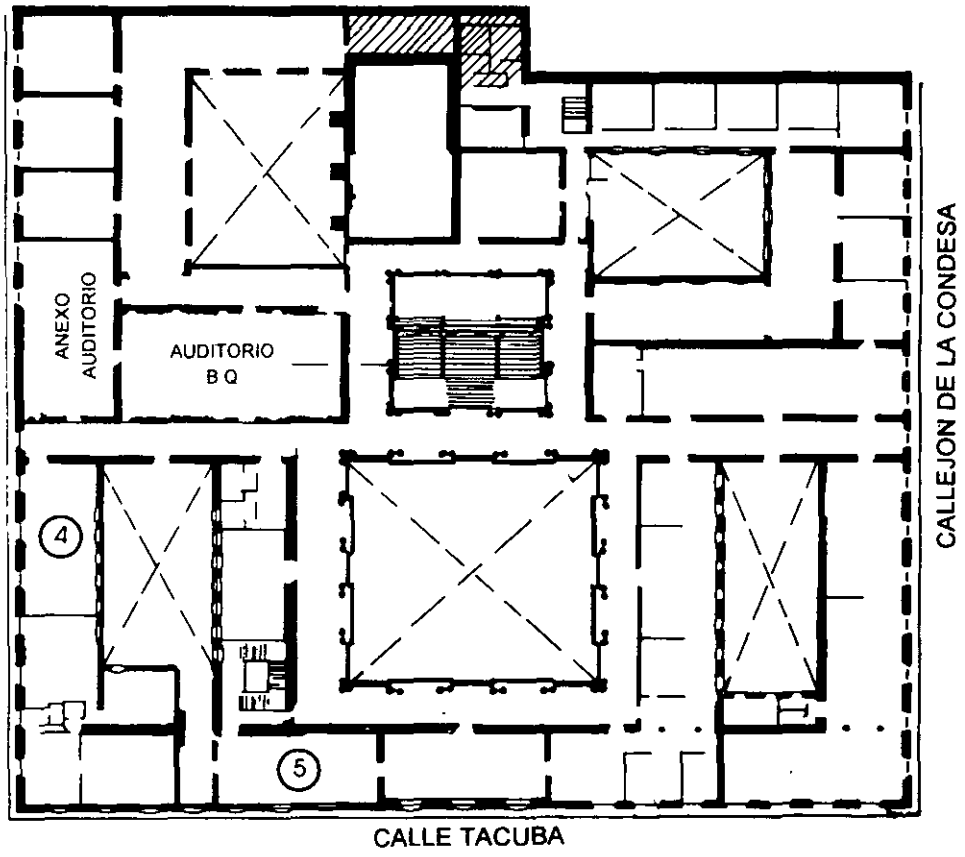
División de Educación Continua.

Palacio de Minería Calle de Tacuba 5 Primer piso Deleg. Cuauhtémoc 06000 México, D.F. APDO. Postal M-2285
Teléfonos: 512-8955 512-5121 521-7335 521-1987 Fax 510-0573 521-4020 AL 26

PALACIO DE MINERIA

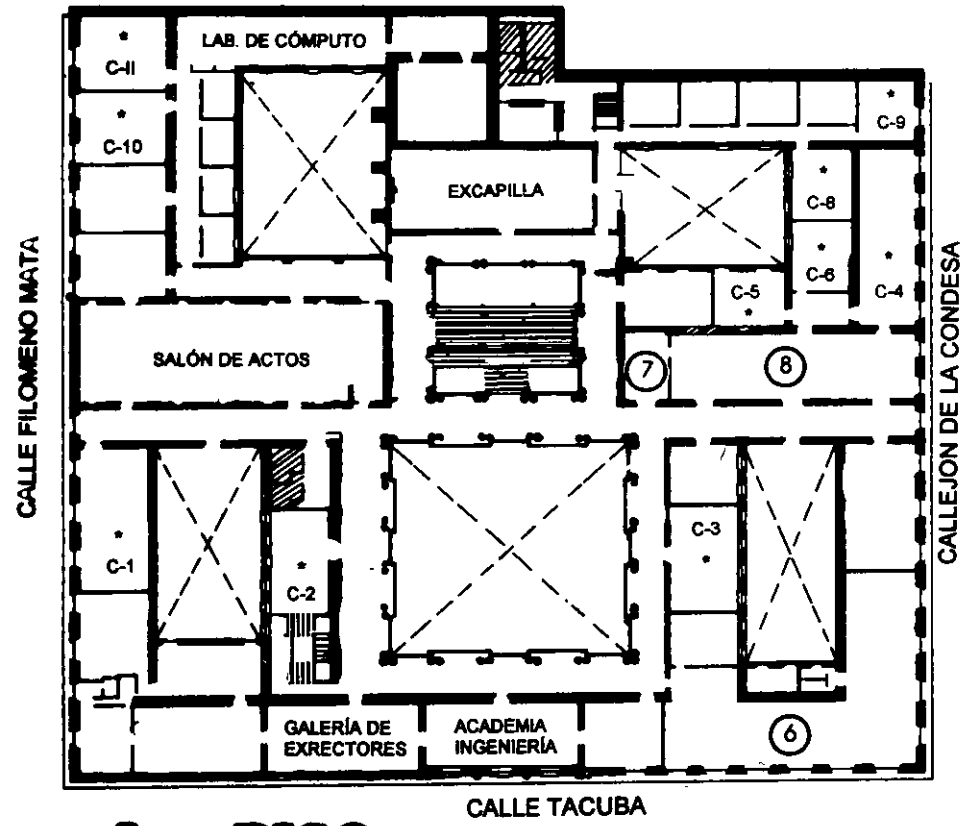


PLANTA BAJA



MEZZANINNE

PALACIO DE MINERIA



Ier. PISO

GUÍA DE LOCALIZACIÓN

1. ACCESO
2. BIBLIOTECA HISTÓRICA
3. LIBRERÍA UNAM
4. CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN "ING. BRUNO MASCANZONI"
5. PROGRAMA DE APOYO A LA TITULACIÓN
6. OFICINAS GENERALES
7. ENTREGA DE MATERIAL Y CONTROL DE ASISTENCIA
8. SALA DE DESCANSO

SANITARIOS

* AULAS

DIVISIÓN DE EDUCACIÓN CONTINUA
FACULTAD DE INGENIERÍA U.N.A.M.
CURSOS ABIERTOS

FACULTAD DE INGENIERIA U.N.A.M.
DIVISION DE EDUCACION CONTINUA

Curso de Seguridad en Redes

Ing. Israel Quiroz Plata

29 de junio de 1998

Palacio de Minería Calle de Tacuba 5 Primer piso Deleg Cuauhtémoc 06000 México, D.F. APDO. Postal M-2285
Teléfonos: 512-8955 512-5121 521-7335 521-1987 Fax 510-0573 521-4020 AL 26

Curso de Seguridad en Redes

Ing. Israel Quiroz Plata

29 de junio de 1998

1 Presentación

1. Quién es el profesor.
2. Qué hace.
3. Cómo localizarlo.
4. Temario.
5. Forma de calificar.
6. Comentarios generales sobre "las reglas del juego".

2 Antecedentes

Bibliografía: [25, 21, 14, 23, 31, 26, 27, 20, 30, 32, 17, 36]

1. ¿Qué es seguridad en cómputo?
2. Por qué vamos a hablar de "seguridad en cómputo" y no de "seguridad en redes".
3. Breve historia de los sistemas operativos.
4. Breve historia de Unix.
5. La seguridad en los sistemas operativos y en Unix.
6. La seguridad y las redes de computadoras.
7. Algunas historias de seguridad.

3 Historia de la Seguridad en Cómputo

Bibliografía: [25, 21]

1. La seguridad en los inicios del cómputo.
2. Cómo surge la necesidad de la seguridad.
3. La Seguridad por Obscuridad.
4. Conceptos modernos de la seguridad.

4 Legislaciones sobre seguridad

Bibliografía: [25, 31, 11, 26]

1. ¿Por qué legislar y regular sobre seguridad?
2. Historia de las leyes relacionadas con seguridad.
3. El Libro Naranja.
4. Otras propuestas.

5 Algunos conceptos teóricos

Bibliografía: [25, 14, 23]

1. Tipos de seguridad.
   (a) Control de acceso.
   (b) Disponibilidad.
   (c) Privacidad.
   (d) Integridad y autenticidad.
   (e) Consistencia.
   (f) Auditoría.
2. Vulnerabilidades.
3. Amenazas.
4. Contramedidas.

6 Riesgos y costos de la seguridad

1. Evaluación de riesgos. ¿Qué tanta seguridad se necesita?
2. ¿Podemos tener seguridad absoluta?
3. Costos
   - Ciclos de reloj.
   - Almacenamiento.
   - Tiempo.
   - Políticas y procedimientos.
   - Infraestructura.
   - Personal.
4. ¿Qué tipos de mecanismos?
   - Software.
   - Hardware.
   - Mixtos.
5. El primer paso.

7 Mecanismos básicos de seguridad

Bibliografía: [21, 18, 6, 9, 7, 14, 37]

1. Control de acceso
   - Mecanismos de autenticación.
   - Permisos de acceso a los recursos.
   - Firewalls.
2. Disponibilidad
   - Respaldos.
   - Espejos.
3. Privacidad.
   - Control de acceso.
   - Cifrado.
4. Integridad, autenticidad y consistencia.

- Listas de verificación.
- Sistemas confiables.
- Cifrado.

5. Auditoría.
    - Bitácoras.
    - Configuraciones correctas.

8 Ataques e incidentes

1. ¿Qué es un incidente?
2. ¿Qué es un ataque?
3. Tipos de incidentes.
    - Intencionales.
    - No intencionales.
    - Físicos y naturales.
4. Tipos de ataques.
    - Internos.
    - Externos.
    - Intencionales.
    - No intencionales.
    - En transmisión de datos.
    - Por emisión de datos.
    - Físicos.

9 Cifrado

1. ¿Qué es el cifrado?
2. ¿Para qué sirve?
    - Privacidad.
    - Integridad.
    - Autenticación.
3. Tipos de sistemas de cifrado.

- De llave privada.
- De llave pública.
- Ejemplos.

4. ¿Qué es la fortaleza de un cifrado?
    - Complejidad.
    - Confidencialidad de algoritmos.
5. Ataques al cifrado.
    - Fuerza bruta.
    - Texto en claro conocido.
    - Texto cifrado conocido.
    - Ataques algorítmicos.
    - Ataques de implementación.
6. Estándares, legislaciones y restricciones.
7. Algunas aplicaciones.
    - PGP
    - S/Key
    - Kerberos
    - Secure RPC

10 Políticas de seguridad

1. ¿Qué son las políticas de seguridad?
2. ¿Por qué son necesarias?
3. Tipos de políticas
    - Uso adecuado.
    - Acceso.
    - Contraseñas.
    - Usuarios y administradores.
    - Licencias y derechos de autor.
    - Acceso a Internet.
4. ¿Cómo elaborar políticas?

11 Administración de seguridad

1. ¿Qué es un oficial de seguridad?
2. ¿Qué obligaciones tiene?
3. ¿De qué privilegios goza?
4. ¿Cuáles son sus limitaciones?
5. Consideraciones éticas.
6. ¿Cómo establecemos la confiablidad?
7. Formas de administración de seguridad.
   - Centralizada.
   - Distribuida.

12 Respuesta a incidentes

1. ¿Qué hacer?
2. Etapas.
   (a) Detección.
   (b) Contención del daño.
   (c) Recuperación.
   (d) Investigación.
3. Recursos útiles.
   - Herramientas de monitoreo y control.
   - Respaldos.
   - Redundancia.
   - Recursos fuera de sitio.
4. ¿A quién acudir?

Referencias

[1] Steven M. Bellovin. There be dragons. In *Proceedings of the Thirs Usenix UNIX Security Symposium*, Murray Hill, NJ, August 15 1992. AT&T Bell Laboratories.

[2] Russell L. Brand. *Coping with the Threat of Computer Security Incidents. A Primer from Prevention through Recovery*, June 8 1990.

[3] Canadian System Security Centre, Communicatons Security Establishment, Government of Canada. *Canadian Trusted Computer Product Evaluation Criteria*, January 1993. Disponible en `http://www.alw.nih.gov/Security/FIRST/papers/criteria/ctcpec1.ps`.

[4] Smoot Carl-Mitchell and John S. Quarterman. Building internet firewalls. *UnixWorld*, pages 93–102, February 1992.

[5] CERT. CERT. *Bridge*, March 1990. Disponible en `ftp://cert.org/pub/cert\_advisories/cert-article`.

[6] D. Brent Chapman and Elizabeth D. Zwicky. *Building Internet Firewalls*. O'Reilly & Associates, Inc., 103 Morris Street, Suite A, Sebastopol, CA 95472, September 1995.

[7] Andrew Cherry, Mark W. Henderson, William K. Nickless, Robert Olson, and Gene Rackow. Pass or fail: A new test for password legitimacy. Technical report, Applied Mathematical Sciences subprogram for the Office of Energy Research, U. S. Department of Energy, September 25 1992.

[8] Bill Cheswick. The design of a secure internet gateway. Technical report, AT&T Bell Laboratories, Murray Hill, New Jersey 07974, September 10 1991. Disponible en `ftp://ftp.super.unam.mx/pub/security/doc/gateway.dvi.gz` y `ftp://ftp.super.unam.mx/pub/security/doc/gateway.ps.gz`.

[9] William R. Cheswick and Steven M. Bellovin. *Firewalls and Internet Security*. Addison-Wesley, Reading, Massachusetts, 1994.

[10] David A. Curry. Improving the security of your unix system. Reporte ITSTD-721-FR-90-21, SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025-3493, April 1990.

[11] Department of Defense. *Department of Defense Trusted Computer System Evaluation Criteria (DoD 5200.28-STD)*, Dec 1985. Disponible en `http://www.pinsight.com:80/~royg/security/dod/orange_book/orange00.htm% 1`.

[12] David Ferbrache and Gavin Shearer. *Unix Installation Security & Integrity*. PTR Prentice Hall, Englewood Cliffs, New Jersey 07632, 1993.

[13] Simson Garfinkel. *PGP—Pretty Good Privacy*. O'Reilly & Associates, Inc., 103 Morris Street, Suite A, Sebastopol, CA 95472, January 1995.

[14] Simson Garfinkel and Gene Spafford. *Practical Unix & Internet Security.* O'Reilly & Associates, Inc., 103 Morris Street, Suite A, Sebastopol, CA 95472, second edition, Abril 1996.

[15] Craig Hunt. *TCP/IP Network Administration.* O'Reilly & Associates, Inc., 103 Morris Street, Suite A, Sebastopol, CA 95472, first edition, January 1994.

[16] *Information Technology Security Evaluation Criteria*, June 1991. Disponible en `http://www.alw.nih.gov/Security/FIRST/papers/criteria/itsec.txt`.

[17] Brian W. Kernighan and Rob Pike. *El Entorno de Programación Unix.* Prentice-Hall Hispanoamericana, 1987.

[18] Gene H. Kim and Eugene H. Spafford. Experiences with tripwire: Using integrity checkers for intrusion detection. Purdue Technical Report CSD-TR-94-012, COAST Laboratory, Department of Computer Sciences, Purdue University, West Lafayette, IN 47907-1398, February 21 1994.

[19] Daniel V. Klein. "foiling the cracker": A survey of, and improvements to, password security. Technical report, Software Engineering Institute, Carnegie Mellon University, 1992.

[20] Robert Morris and Ken Thompson. Password security: A case history. *Communications of the ACM*, 22(11):594–597, November 1979.

[21] Alec Muffet et al. Faq: Computer security frequently asked questions. Disponible en los grupos de Usenet comp.security.misc y alt.security, Dec 1993. Versión 2.2.

[22] R. M. Needham and M. D. Schroeder. Using encryption for authentication in large networks of computers. *Communications of the ACM*, 21(12):993–999, December 1978.

[23] National Institute of Standards and Technology. Selected bibliography of key computer security literature. Disponible en `ftp.super.unam.mx:/pub/security/lit/800-1complete.txt.gz`., 1980–1989.

[24] John S. Quarterman and Smoot Carl-Mitchell. Tutorial: Local protection for networked systems. *UnixWorld*, X(7):64–72, July 1993.

[25] Deborah Russel and G. T. Gangemi Sr. *Computer Security Basics.* O'Reilly & Associates, Inc., 103 Morris Street, Suite A, Sebastopol, CA 95472, July 1992.

[26] Tsutomu Shimomura and John Markoff. *Takedown: The pursuit and capture of Kevin Mitnick, America's most wanted computer outlaw—by the man who did it.* Hyperion, New York, 1996.

[27] Eugene H. Spafford. The internet worm incident. Technical Report CSD-TR-933, Department of Computer Sciences, Purdue University, West Lafayette, IN USA 47907-2004, September 19 1991.

[28] M. St. Johns. Identification protocol. Request for Comments 1413; Network Working Group, US Department of Defense, February 1993. Disponible en `ftp://nic.ddn.mil/rfc/rfc1413.txt`.

[29] Hal Stern. *Managing NFS and NIS*. O'Reilly & Associates, Inc., 103 Morris Street, Suite A, Sebastopol, CA 95472, first edition, June 1991.

[30] Clifford Stoll. Stalking the wily hacker. *Communications of the ACM*, 31(5):484–497, May 1988.

[31] Clifford Stoll. *The Cuckoo's Egg: Tracing a Spy Through the Maze of Computer Espionage*. Doubleday, New York (NY), 1989.

[32] Andrew S. Tanenbaum. *Sistemas Operativos: Diseño e Implementación*. Prentice-Hall Hispanoamericana, Mexico, 1988.

[33] U. S. National Institute for Standards and Technology. *Common Criteria*, January 1996. Disponible en `http://www.tno.nl/instit/fel/refs/cc.html`.

[34] Wietse Venema and Dan Farmer. Improving the security of you site by breaking into it. Disponible en `http://www.super.unam.mx/seguridad/docs/admin-guide-to-cracking.101.ht% ml`.

[35] Larry Wall and Randal L. Schwartz. *Programming Perl*. O'Reilly & Associates, Inc., 103 Morris Street, Suite A, Sebastopol, CA 95472, first edition, March 1992.

[36] Diego Zamboni. *Ingeniero en Computación*. PhD thesis, Universidad Nacional Autónoma de México, 1995. Disponible en `ftp://ftp.super.unam.mx/pub/security/doc/tesis/Diego_Zamboni.ps.gz`.

[37] Diego M. Zamboni. Comentarios sobre configuración de reglas en passwd+. Publicado originalmente en mensajes distribuidos en la lista de correo electrónico gasu, 1994. Disponible por FTP anónimo en `ftp.super.unam.mx:/pub/security/doc/passwd+.comentarios.gz`.

[38] Diego M. Zamboni. TCP-Wrapper: Introducción, instalación y uso. Publicado originalmente en mensajes distribuidos en la lista de correo electrónico gasu, 1994. Disponible por FTP anónimo en `ftp.super.unam.mx:/pub/security/doc/tcp\_wrapper.tutorial.gz`.

[39] Philip Zimmermann. *PGP User's Guide*. Phil's Pretty Good Software, Oct 1994. Disponible por FTP anónimo en `ftp://ftp.super.unam.mx/pub/security/tools/PGP/`.

Seguridad en Redes de Computadoras

Ing. Israel Quiroz Plata
Area de Seguridad en Cómputo
Dirección General de Servicios de Cómputo Académico, UNAM.
asc@ds5000.super.unam.mx

El instructor

- Israel Quiroz Plata
- Jefe del Area de Seguridad en Cómputo, DGSCA, UNAM.
- Cómo localizarme: israel@conga.super.unam.mx

Forma de calificar

Tareas:	30%
Examen:	70%
Asistencias	(min. 80%)
Participación en clase	

Reglas del juego

- Las tareas se entregan por correo electrónico (curso@conga.super.unam.mx).
- Revisar que los mensajes se envíen correctamente (Cc:)
- Se recibirán tareas retrasadas hasta 24 horas, y se calificarán sobre 8.

¿Qué es Seguridad en Cómputo?

Términos como "seguridad", "protección" y "privacidad" pueden tener más de un significado, dependiendo de quién lo aplica, y en qué ámbito se aplica.
Incluso los profesionales que trabajan en el área de seguridad no siempre coinciden en lo que estos términos significan.

¿Qué es Seguridad en Cómputo?

Una definición bastante práctica de seguridad es:

> "Un sistema es seguro si se puede *confiar* en que él y su software se comporten como los usuarios esperan que lo hagan."

¿ "Seguridad en Cómputo" o "Seguridad en Redes"?

La Seguridad en Redes es una parte muy importante de la Seguridad en Cómputo, y hay quienes la consideran como área separada.
Sin embargo, es vital tener el conocimiento global, porque una no puede existir sin la otra.

Seguridad en Cómputo

Ing. Israel Quiroz Plata
Area de Seguridad en Cómputo
Dirección General de Servicios de Cómputo Académico, UNAM.
asc@ds5000.super.unam.mx

(Muy breve) historia de los Sistemas Operativos

Pre-1965: Operación mecánica y manual.

1965-1980: Primeros sistemas operativos modernos. OS/360 (IBM), CTSS (MIT), MULTICS (AT&T, GE, MIT), Unix (AT&T).

Post-1980: Sistemas personales. CP/M, MS-DOS, Unix, Macintosh.

Breve historia de Unix

1960's: AT&T, Honeywell, General Electric y el MIT inician proyecto de información distribuida llamado *Multics*.

1969: AT&T decide salirse del proyecto.

Ken Thompson y Dennis Ritchie, de AT&T, continuaron con el desarrollo por su cuenta, renombrándolo *Unix*.

1973: Thompson reescribe Unix en lenguaje C. La Universidad de California en Berkeley ordena una copia del sistema.

Breve historia de Unix

1977: Berkeley comienza a distribuir su versión de Unix (*Berkeley Software Distribution* o BSD).

Actualmente: Unix se utiliza en millones de computadoras de todo tipo en todo el mundo, y en todos los ámbitos.

La seguridad en los sistemas operativos

- De los sistemas personales mejor ni hablamos.
- Los problemas reales se presentan en los sistemas multiusuario y multitarea, como Unix.

Unix y la seguridad

> *"No fue diseñado desde el principio para ser seguro. Fue diseñado con las características necesarias para poder darle mantenimiento a la seguridad."*
>
> –Dennis Ritchie

Debido a las muchas "historias de terror" existentes sobre problemas de seguridad de Unix, mucha gente tiene la convicción de que la frase "Seguridad en Unix" es un oxímoron.

Unix y la seguridad

Sin embargo, la reputación de Unix como sistema inseguro no viene de su diseño sino de sus implementaciones. Unix nació en ambientes donde la seguridad no era una preocupación.

Por esto, los vendedores de Unix han sido lentos en incorporar mecanismos de seguridad en sus sistemas.

Las redes de computadoras y la seguridad

Hoy en día las redes de computadoras conectan entre sí miles de computadoras en todo el mundo, haciendo posible el acceso a un sistema desde cualquier lugar del planeta. Esto es particularmente cierto en redes de alcance mundial como *Internet*, y convierten a la seguridad en algo que debe preocupar a todos los usuarios, y ya no solo a los administradores.

Algunas historias de seguridad

1987–1988: Cliff Stoll rastrea un incidente internacional.

1988: El Gusano de Internet.

1993: Cray de la UNAM.

1994–1995: Rastreo y captura de Kevin Mitnick.

La seguridad en los inicios del cómputo

- Control físico.
- Parte de la seguridad de la planta.
- Personal altamente capacitado y normalmente confiable.

Pero ahora...

- Amplísima difusión y acceso a equipo de cómputo.
- Conectividad "total".

¿Cómo surge la necesidad de la seguridad?

- Está "de moda", pero no se nuevo.
- Surge como parte de los controles de la información.
- Se hace más necesaria con la popularización de las tecnologías.

Seguridad por obscuridad

Surge como uno de los primeros conceptos en seguridad en cómputo, pues se toma directamente de otros ámbitos, como el militar.
Se deriva del concepto de la "necesidad de saber". La información es dividida, y a cada quien solo se le proporciona lo que necesita para hacer su trabajo.

Ventajas de la Seguridad por obscuridad

- Permite proteger partes confidenciales de la información.
- Impide la obtención de información por inferencia.

Desventajas de la Seguridad por obscuridad

- No sirve si la información que se está ocultando es averiguable por otras fuentes.
- Impide que incluso los usuarios legítimos tengan acceso a la información que necesitan.
- Reduce el número de personas preparadas para reaccionar adecuadamente ante una emergencia.

Idea moderna de la seguridad

- Seguridad algorítmica.
- Mecanismos y algoritmos lo más públicos posibles, pero inherentemente seguros.
- OJO: Hay información que **debe** ser confidencial.

¿Por qué reglamentar sobre seguridad?

- Tener lineamientos objetivos y específicos al respecto.
- Tener puntos de comparación y antecedentes tecnológicos, civiles y jurídicos.

¿Por qué legislar sobre seguridad?

- Los gobiernos quieren tener control sobre todo.
- Los gobiernos necesitan tener control sobre algunas cosas.
- A todos nos benefician ciertos controles, lineamientos y garantías legales.

Reglamentaciones sobre seguridad

- En E. U., hay trabajos desde 1950, y de forma "moderna" desde 1968.
- En E. U. ya se han ejercido acciones legales por crímenes de seguridad en cómputo.
- En algunos otros países también hay trabajo al respecto.
- En México no hay nada.

Problemas con las reglamentaciones y legislaciones

- La tecnología avanza demasiado rápido.
- No siempre se tienen en cuenta las necesidades y opiniones de quienes usan la tecnología.
- Muchas veces las reglas son hechas por gente que no sabe mucho.

El Libro Naranja

Llamado oficialmente *Department of Defense Trusted Computer System Evaluation Criteria*, este documento establece los requerimientos que debe cumplir un sistema para poder ser calificado formalmente como confiable.
El concepto central del Libro Naranja es que es posible medir la confianza que se pone en un sistema.
El Libro Naranja define cuatro divisiones jerárquicas de seguridad, y cada una de ellas se divide en clases.

Clasificación según el Libro Naranja

División	Clase	Descripción
D		Protección mínima
C		Protección discrecional
	C1	Protección discrecional.
	C2	Protección de acceso controlado.
B		Protección obligatoria.
	B1	Protección por etiquetas.
	B2	Protección estructurada.
	B3	Dominios de seguridad.
A		Protección verificada.
	A1	Diseño verificado.

Comentarios sobre el Libro Naranja

- Sigue siendo el estándar con el que se mide la seguridad.
- Puede ser "demasiado formal" para ciertas aplicaciones.
- Fue escrito hace ya 11 años.

Otras propuestas

ITSEC: (*Trusted Information Technology Security Evaluation Criteria*), publicado en Alemania en 1992.

FC-ITS: (*Federal Criteria for Information Technology Security*), E. U., 1992 (se le suele llamar "el nuevo Libro Naranja").

CTCPEC: (*Canadian Trusted Computer Product Evaluation Criteria*), Canadá, 1993.

CC: (*Common Criteria*), E. U. Es el esfuerzo por unificar todos los demás documentos.

Tipos de seguridad

Existen múltiples tipos de seguridad en cómputo. Como usuarios y como administradores, es necesario conocer al menos los tipos básicos, para decidir cuáles son los más importantes para nosotros.

Privacidad

Consiste en proteger la información contra ser leída por nadie que no tenga autorización explícita para hacerlo.
Incluye no solo proteger la información en su totalidad, sino también las piezas individuales de información que puedan ser utilizadas para inferir otros elementos de información confidencial.
No confundir con "Seguridad por Obscuridad".

Integridad de datos

Proteger la información contra ser modificada sin el permiso de su dueño.
La información a ser protegida incluye no solo la que está almacenada directamente en los sistemas de cómputo, sino elementos menos obvios, como respaldos, documentación, registros de contabilidad del sistema, etc.

Disponibilidad

Proteger los servicios de cómputo de manera que no se degraden o dejen de estar disponibles a los usuarios de forma no autorizada.

Consistencia

Es asegurar que el sistema siempre se comporte de la forma esperada.

Regulación de acceso

Consiste en controlar quién utiliza el sistema o cualquiera de los recursos que ofrece.

Auditoría

Consiste en contar con los mecanismos para poder determinar qué es lo que sucede en el sistema, qué es lo que hace cada uno de los usuarios, y los tiempos y fechas de dichas acciones.

Vulnerabilidades

Son los puntos débiles de un sistema de cómputo, a través de los cuales su seguridad se puede ver afectada.

Vulnerabilidades

- Físicas.
- Naturales.
- Hardware y Software.
- Medios de almacenamiento.
- Emanaciones.
- Comunicaciones.
- Humanas.

Amenazas

Son los elementos que, a través de alguna vulnerabilidad del sistema, pueden causar un daño en el sistema.

Amenazas

- Naturales y físicas.
- No intencionales.
- Intencionales.

Amenazas intencionales

- Externos.
- Internos.

Contramedidas

Son las acciones que se pueden tomar para prevenir o minimizar los daños.

Contramedidas

- Seguridad interna del sistema.
- Seguridad en comunicaciones.
- Seguridad física.
- Seguridad humana.

Análisis de riesgos

Aunque todos los tipos de seguridad son importantes, sus prioridades varían de una organización a otra. Diferentes ambientes tienen diferentes preocupaciones de seguridad, y por lo tanto deben establecer sus políticas y mecanismos de acuerdo a ellos.
El administrador de seguridad necesita entender perfectamente las necesidades del ambiente en el que trabaja, para definir los procedimientos adecuadamente.

¿Qué es un análisis de riesgos?

- Cuantificar el impacto de las amenazas potenciales en un sitio de cómputo.
- Ayuda a balancear los riesgos contra los costos.
- Ayuda a decidir qué riesgos prevenir, limitar o aceptar.
- Ayuda a obtener apoyo directivo.

Elementos de un análisis de riesgos

- Identificar y valuar nuestras "propiedades".
- Identificar las amenazas a esas propiedades y sus posibles costos.
- Identificar los controles de seguridad que se tienen.
- Identificar sus debilidades.

¿Quién realiza en análisis de riesgos?

Lo mejor es realizarlo con un grupo de personas que son responsables de distintas áreas de soporte en la organización.
Si lo hace una sola persona, se corre el riesgo de generar graves inconformidades.

Pasos de un análisis de riesgos

- Recolectar la información acerca de propiedades, amenazas y controles de seguridad.
- Evaluar los resultados para encontrar áreas débiles.
- Hacer recomendaciones de los cambios que se deben hacer.
- Escribir el reporte formal y someterlo a revisión y aprobación oficial.

¿Formal o informal?

- Es importante determinar si se quiere hacer un análisis formal o informal.
- Un análisis formal debe incluir todas las posibles propiedades, riesgos y controles de seguridad.
- Un análisis informal puede incluir solamente propiedades importantes, las amenazas más comunes y los controles de seguridad más relevantes.
- El tipo de análisis a realizar debe determinarse con base en los objetivos y tamaño de la organización.

Recolección de información

- Evaluar propiedades y amenazas.
- Evaluar protecciones físicas.
- Evaluar sistemas de soporte (aire, potencia, refrigeración, etc.)
- Evaluar mecanismos de protección a datos sensitivos.
- Evaluar mecanismos de protección contra errores y fraude.
- Estimar la confiabilidad del hardware.
- Identificar riesgos asociados con el personal.
- Evaluar políticas y procedimientos existentes.

Clasificación de propiedades

- Propiedad intelectual (programas, datos, documentación, páginas de WWW, etc.)
- Propiedad física (computadoras, medios de almacenamiento, equipo de redes).
- Procesos y servicios de cómputo (tiempo de CPU, funciones del personal de soporte).

Estimados de costo de propiedades

- Cada propiedad evaluada debe tener un costo asociado.
- El costo de la propiedad física es el costo de reemplazo.
- El costo de la propiedad intelectual incluye cómo la organización reaccionaría si los datos se perdieran o comprometieran de manera total (¿cómo recuperar los datos? ¿se pueden recuperar?).
- El costo de los recursos de cómputo puede ser el costo de hacer funcionar los recursos por un período de tiempo.

Costo de propiedades de software

- Un método es estimar el tiempo requerido para recompilar, probar, configurar y reinstalar el software perdido.
- Utilizar estimados para el peor caso.
- Utilizar números redondos (1 semana, 2 semanas, 1 mes, etc.)
- Normalizar (e.g. para paquetes simples el tiempo es una semana, para paquetes complejos es 3 semanas).

Etapa de evaluación de riesgos

- Etapas:
  - Identificar todos los posibles riesgos.
  - Determinar la frecuencia potencial de cada amenaza.
  - Estimar la pérdida en dólares de cada pérdida.
- Un análisis formal incluye todos los riesgos posibles, y uno informal incluye solo los de mayor frecuencia.

Frecuencia Anual Estimada por riesgo

- A cada riesgo hay que asignarle una Frecuencia Anual Estimada (FAE).
- La FAE es el número estimado de ocurrencias de la amenaza en un período de un año.

Frecuencia Anual Estimada por riesgo

- Para facilidad en los cálculos, redondear a múltiplos de 10:
  - Una vez en 300 años.
  - Una vez en 30 años.
  - Una vez en 3 años (1000 días).
  - Una vez en 100 días.
  - Una vez en 10 días.
  - Una vez por día.
  - 10 o 100 veces por día.

Indice de FAE

- Para reducir la complejidad de los cálculos, se puede usar un índice FAE:
  - 1=Una vez en 300 años.
  - 2=Una vez en 30 años.
  - 3=Una vez en 3 años.
  - 4=Una vez en 100 días.
  - 5=Una vez en 10 días.
  - 6=Una vez al día.

Pérdida Anual Esperada

- La pérdida anual esperada (PAE) asociada a un riesgo es el costo de la ocurrencia de un riesgo multiplicado por la FAE de ese riesgo:

$$PAE = Impacto \times FAE$$

- Para cada riesgo, calcular las pérdidas si ocurriera.

Pérdida Anual Esperada

- Para simplificar los cálculos, utilizar números redondos ($100, $1,000, $10,000, etc.)
- Para amenazas humanas como accesos no autorizados, el impacto es el costo de las horas–hombre requeridas para investigar el incidente.
- Para amenazas humanas accidentales, la PAE puede ser una medida de la pérdida de tiempo de cómputo.

Pérdida Anual Esperada

- Para amenazas ambientales, la PAE puede ser un estimado del costo de los daños y el tiempo de procesamiento perdido.
- Para amenazas como fallas de hardware, la PAE se puede basar en el costo estimado de operar la máquina durante el período que no funcione.

Identificar los controles de seguridad

- La siguiente fase es identificar los controles y protecciones que ya estén funcionando.
- Tipos de protecciones:
  - Físicas.
  - Administrativas.
  - Técnicas.
- Una protección debe impedir o limitar el daño producido por una amenaza.
- Las protecciones tienen que ser efectivas en costo.

Protecciones físicas

- Controles de acceso a edificios.
  - Tarjetas o llaves de acceso.
  - Video cámaras en los accesos.
- Controles internos.
  - Puertas con cerraduras.
  - Video cámaras en los pasillos.
  - Cuartos de cómputo cerrados.
  - Puestos de vigilancia.

Protecciones físicas

- Protecciones contra desastres naturales.
  - Detectores de fuego, calor y humo.
  - Extingidores.
  - Detectores sísmicos.
- Monitores ambientales.
  - Monitores y controles de temperatura.
  - Monitores de humedad.
- Controles contra robo.

Protecciones administrativas

- Políticas y procedimientos formales.
  - Política de Cuentas y de Uso Aceptable.
  - Política de Seguridad del Sitio y procedimiento de respuesta a incidentes.
  - Procedimiento de terminación de empleados.

Protecciones administrativas

- Entrenamiento.
  - Orientación a nuevos empleados.
  - Concientización de seguridad.
  - Entrenamiento de respuesta a incidentes.
- Otros.
  - Mensajes de inicio de sesión.
  - Controles para información confidencial.

Protecciones técnicas

- Monitoreo y control de acceso a servicios.
- Controles de acceso en archivos.
- Herramientas de monitoreo de sistemas.
- Firewalls.
- Controles de acceso a la cuenta de root.

Cómo identificar las áreas débiles

- Probar las protecciones que se tienen.
- ¿Hay algunas que no estén funcionando bien?
- ¿Hay amenazas para las que no hay protección?
- Evaluar todos los aspectos de seguridad del sitio.

Cómo probar protecciones físicas

- Invitar a alguien a pasearse por las instalaciones, y ver hasta dónde puede entrar.

Preguntas al respecto

- ¿Los controles de acceso son suficientes para impedir la entrada de extraños?
- ¿Los empleados detienen y cuestionan a los extraños?
- ¿Las áreas restringidas son realmente restringidas?
- ¿Los empleados tienen identificaciones apropiadas?

Cómo probar protecciones en procedimientos

- Hacer una simulación de incidente.
- Intentar hacer ingeniería social.

Preguntas al respecto

- ¿Hay un equipo de respuesta a incidentes?
- ¿Hay un procedimiento de respuesta a incidentes?
- ¿El equipo de respuesta entiende su papel?
- ¿ "Atención a usuarios" sabe a quién llamar ante un problema de seguridad?
- ¿Están bien establecidas las líneas de autoridad para ciertas peticiones y tareas?

Cómo probar las protecciones técnicas

- Intentar entrar en nuestro propio sitio, o decirle a alguien que lo intente.
- Hacer que alguien de fuera lance un ataque masivo contra la red.
- Hacer cambios no autorizados en uno o más sistemas y ver si alguien se da cuenta.
- Crear una cuenta falsa y ver si alguien se da cuenta y qué hace.
- Leer "Improving the security of your site by breaking into it".

¿Podemos eliminar completamente los riesgos?

"El único sistema totalmente seguro es aquel que está apagado, desconectado, guardado en una caja fuerte de titanio, encerrado en un bunker *de concreto, rodeado por gas venenoso y cuidado por guardias armados muy bien pagados. Aún así, no apostaría mi vida por él."*

–Eugene Spafford

Costos de la seguridad

Se dice que la seguridad absoluta solo se puede obtener a un costo infinito.

Costos de la seguridad

- Ciclos de reloj.
- Almacenamiento.
- Tiempo.
- Políticas y procedimientos.
- Infraestructura.
- Personal.

¿Qué tipos de mecanismos utilizar?

- Software.
- Hardware.
- Mixtos.

El primer paso

Responder a las siguientes preguntas:

- ¿Qué se quiere proteger?
- ¿Contra qué se quiere proteger?
- ¿Cuánto tiempo, dinero y esfuerzo se está dispuesto a invertir para protegerlo?

Mecanismos básicos de seguridad

Existen técnicas y mecanismos que nos ayudan a obtener los distintos tipos de seguridad mencionados.
Estos mecanismos deben aplicarse de acuerdo al análisis de riesgos y costos que se haya realizado.
Vamos a discutir mecanismos que permiten obtener cada uno de los tipos de seguridad mencionados.

Control de acceso

- **Objetivos:**
  1. Mantener fuera del sistema a quienes no tienen derecho a utilizarlo.
  2. Impedir que utilicen los recursos y servicios personas que no tengan derecho a ellos.
- **Técnicas:**
  1. Mecanismos de autenticación.
  2. Permisos de acceso a los recursos.
  3. Firewalls.

Autenticación

El usuario debe decirle al sistema quién es, y *demostrarlo*. A esto se le llama *autenticación*.

Formas "clásicas" de autenticación

1. Decirle a la computadora algo que uno sabe (una contraseña).
2. Mostrarle a la computadora algo que uno tiene (una tarjeta inteligente).
3. Dejar que la computadora mida algo sobre uno (la huella digital).

Ninguno de estos sistemas es infalible.

Passwords (contraseñas)

El método más utilizado es el primero, por ser el más fácil de implementar y el más barato. Al secreto que el usuario comparte sólo con la computadora se le llama *password*.
Casi siempre, aunque se utilice algún otro mecanismo de autenticación, representa un complemento, y no un sustituto, de un *password*.
Desafortunadamente, también es el más fácil de violar.

¿Por qué usar passwords?

Por la misma razón que cerramos con llave la puerta de nuestra casa. No queremos que cualquiera que pase por enfrente pueda abrir la puerta y entrar.
En el mundo actual, con las computadoras conectadas en redes de amplia cobertura, hay muchas personas que se dedican a probar las "puertas" de todos los sistemas que puedan encontrar. Si esta puerta está abierta, un vándalo puede entrar sin problema y causar daño.

Importancia de los passwords

Los passwords son la primera línea de defensa contra accesos no autorizados.
Aunque es posible entrar a un sistema o robar información sin tener que entrar en una cuenta primero, la gran mayoría de las intromisiones se deben a passwords mal elegidos o mal protegidos.

> *Un password debe ser como un cepillo de dientes: úsalo diariamente, cámbialo de vez en cuando, y **NO** lo compartas con los amigos".*

Passwords débiles

Un password *débil* o *malo* es uno que es fácilmente adivinable.

Ejemplos de passwords débiles

Nombres (del usuario, de su esposa, su mascota, su hijo, etc.)
El nombre del sistema operativo que se está usando.
El nombre de la máquina que se está usando.
Información que sea fácilmente obtenible acerca del usuario (cumpleaños, teléfono, calle, etc.)
Palabras que aparezca en un diccionario.
Nombres de lugares.
Letras repetidas.
Patrones del teclado (como **qwerty**).
Alguno de los anteriores escrito al revés.
Alguno de los anteriores seguido de un solo dígito.

Buenos passwords

Un password *bueno* es aquel que es difícil de adivinar.

Características de los buenos passwords

- Tienen letras mayúsculas y minúsculas.
- Tienen dígitos y/o signos de puntuación así como letras.
- Son fáciles de recordar.
- Son de 7 u 8 caracteres de largo.
- Pueden ser tecleados rápidamente.

Sugerencias para buenos passwords

- Combinar palabras cortas.
- Usar acrónimos.
- Si se usa el mismo password en varias máquinas, variarlo ligeramente en cada una de ellas.
- No escribir los passwords.

Si se debe escribir el password...

- No identificarlo como tal.
- No escribir a qué máquina o cuenta pertenece.
- No pegar el papelito en la máquina.
- Mezclar caracteres aleatorios.
- **Nunca** mandarlo por correo electrónico.

Técnicas administrativas de manejo de passwords

Aunque la mayor parte de la responsabilidad acerca de los passwords recae sobre los usuarios, el administrador de un sistema de cómputo puede tomar ciertas medidas para limitar la vulnerabilidad de los passwords del sistema.

Asignación de passwords

Consiste en que el administrador asigne los passwords de los usuarios, y no les permita cambiarlos.

Asignación de passwords

- Ventaja:
  - Impide que los usuarios elijan passwords débiles.
- Desventajas:
  - Incomoda a los usuarios.
  - Probablemente los usuarios escriban los passwords para recordarlos.

Romper los passwords del sistema

Consiste en utilizar programas que tratan de adivinar los passwords de los usuarios, basándose en información sobre el usuario y sobre su cuenta, así como en diccionarios.
Crack: `ftp://ftp.super.unam.mx/pub/security/tools/crack.tar.gz`

Romper los passwords del sistema

- Ventaja:
  - Permite detectar passwords débiles.
- Desventajas:
  - La detección se hace después de asignado el password.
  - El mismo programa, o los reportes generados, pueden ser usados por los intrusos.
  - Puede ser muy costoso en tiempo de procesamiento.
  - Puede no detectar todos los passwords débiles.

Usar passwords *shadow*

Consiste en no almacenar los passwords cifrados en `/etc/passwd`, sino en otro archivo (`/etc/shadow`), que no sea legible para todos los usuarios.
Shadow: `ftp://ftp.super.unam.mx/pub/security/tools/shadow.tar.gz`

Usar envejecimiento y expiración de passwords

Consiste en que el sistema automáticamente le pida al usuario que cambie su password después de un cierto tiempo.
Es importante encontrar el tiempo adecuado para solicitar el cambio de password, que no sea demasiado corto ni demasiado largo.
Shadow: `ftp://ftp.super.unam.mx/pub/security/tools/shadow.tar.gz`

Desactivar accesos que no serán utilizados

Cuando un usuario no va a utilizar sus permisos de acceso durante un período de tiempo, es prudente desactivar dichos permisos para que no sean utilizados por alguien no autorizado.

Usar programas que impidan elegir passwords débiles

Esta es una de las mejores técnicas. Se trata de reemplazar el mecanismo estándar de selección de passwords con un equivalente, pero que no permita al usuario poner en su cuenta un password débil.
Passwd+: `ftp://ftp.super.unam.mx/pub/security/tools/passwd+.tar.gz`

- Ventaja: evita los passwords débiles antes de que sean establecidos.
- Desventajas:
  - Puede no detectar todos los passwords débiles.
  - Después de instalar el programa es necesario obligar a todos los usuarios a cambiar su password.

Educar a los usuarios

- Ventajas: todas.
- Desventajas: muy difícil de llevar a cabo, y sobre todo de mantener en funcionamiento constante.

Permisos de acceso a los recursos

Distintos sistemas de cómputo tienen distintos mecanismos para especificar qué usuarios pueden acceder a qué recursos, y qué tipo de acceso pueden realizar.
Es muy importante utilizar estos mecanismos de manera apropiada para asegurar la correcta utilización de los recursos de cómputo.

Permisos de acceso

La gran mayoría de los sistemas soportan tres tipos de acceso:

Lectura: Permiso para examinar el contenido de un archivo.

Escritura: Permiso para modificar o reemplazar el contenido de un archivo.

Ejecución: Si se trata de un programa, permiso para ejecutarlo.

Control de acceso por dueño de los recursos

Si cada recurso tiene "dueños" asignados, podemos controlar no solamente **cómo** se accede al recurso, sino también **quién** lo hace.
Pueden existir distintos tipos de "posesión" de un recurso (por ejemplo, puede tener un dueño y un grupo, como en Unix).
Basándose en los dueños de los recursos, se pueden establecer distintos esquemas de control de acceso.

Esquema de control de acceso por tipos de archivos

(utilizado en el sistema operativo Wang SVS/OS CAP 1.0)

Público: Cualquier puede leer o escribir al archivo.

Solo-ejecución: Cualquiera puede ejecutar el archivo. Solo su dueño y el administrador puede leerlo o escribir a él.

Solo-lectura: Cualquier puede leer o ejecutar el archivo. Solo su dueño y el administrador pueden modificarlo.

Privado: Solo el dueño y el administrador pueden leer, escribir o ejecutar un archivo.

Configurable: Esquemas de acceso configurables, normalmente por clases de archivos y usuarios.

Control de acceso por dueño/grupo/público

(utilizado en Unix)
Existen tres categorías de usuarios, que pueden tener diferentes privilegios de acceso sobre los recursos:

Dueño: El creador o dueño del archivo.

Grupo: Un conjunto de usuarios definido por el administrador.

Los demás: Todos los que no son el dueño y no pertenecen al grupo.

Listas de control de acceso

Con listas de control de acceso, se puede especificar, usuario por usuario o grupo por grupo, quién tiene qué privilegios de acceso sobre cada uno de los recursos.
Ejemplo:

```
<john.acct, r>
<jane.pay, rw>
<ken.eng, x>
```

Controles de acceso obligatorios (*Mandatory Access Control*, MAC)

En este esquema, el *sistema* asigna los privilegios de acceso; los usuarios no pueden decidir quién tiene acceso a los recursos.
Se le asignan etiquetas de seguridad a todos los sujetos (usuarios, programas, procesos) y a todos los objetos (archivos, directorios, dispositivos, ventanas) en el sistema.
La etiqueta de un sujeto especifica sobre qué objetos tendrá acceso, y qué tipo de acceso será.
Se le conoce también como *Seguriad multinivel*.

Seguridad multinivel

Cada sujeto y objeto tiene una etiqueta de seguridad, que está compuesta por dos elementos:

Una clasificación que es un nivel jerárquico al que pertenece la etiqueta.

Un conjunto de categorías o compartimientos, que son una clasificación no jerárquica que representa distintas áreas de información dentro de un mismo nivel.

Ejemplo de etiqueta:

```
SECRETO [INGENIERIA VENTAS RELACIONES]
```

Seguridad multinivel

La idea es que incluso alguien que tiene el mayor nivel jerárquico no tiene automáticamente permiso de acceder a toda la información.
Existen conjuntos de reglas bien definidos para saber, a partir de las etiquetas de un sujeto y un objeto, los tipos de acceso que se pueden tener.

Reglas de acceso en seguridad multinivel

No read up: Para lectura, la etiqueta del sujeto debe *dominar* a la del objeto.

No write down: Para escritura, la etiqueta del objeto debe dominar a la del sujeto.

La etiqueta del objeto **A** *domina* a la de **B** si el nivel de A es mayor que el de B y el conjunto de compartimientos de A incluye completamente al de B.

¿Qué es un *Firewall*?

En construcción, es una pared resistente al fuego, que impide que un incendio se propage a todo el edificio.
En redes, es un mecanismo que aisla una red del exterior, controlando muy estrictamente el paso de información hacia y desde la red.

Firewalls internos

Consiste en hacer las subredes internas lo más independientes posible, haciendo que se comuniquen entre ellas a través de ruteadores.

Firewalls externos

Son máquinas que separan la red local del exterior. Toda la comunicación tiene que pasar a través del *firewall*, y éste puede ser configurado para permitir solamente la utilización de ciertos servicios.

Conceptos de arquitectura de un *firewall*

- Filtro (*choke*).
- Compuertas (*gates*).
- Proxies.

El filtro (*choke*)

Una computadora o dispositivo de comunicaciones que restringe el flujo libre de paquetes entre las redes.
Normalmente se trata de un enrutador, pero no tiene que serlo. Puede ser, por ejemplo, una máquina Unix con dos interfases de red.

Las compuertas (*gates*)

Son programas, dispositivos o computadoras dentro del perímetro del *firewall*, y designados especialmente para recibir conexiones desde redes externas y manejarlas de manera apropiada.
En términos generales, su función es asegurar la validez de los paquetes que lleguen, y dejar pasar hacia su destino final solamente los que sean válidos.
La compuerta es el único sistema al que se tiene acceso desde el exterior de la red. Por eso, debe tener activados todos los mecanismos de seguridad pertinentes, y no debe tener cuentas de usuarios.

Proxies

Un proxy es un programa que finge ser otro. En el caso de un *firewall*, un proxy es un programa que recibe peticiones de servicios desde el interior del *firewall*, y las pasa de manera apropiada al exterior.
Los proxies son el mecanismo para proporcionar acceso a Internet a máquinas que estén dentro del *firewall*.

Principales arquitecturas de *firewalls*

- Máquina con dos puertos.
- Filtrado de paquetes.
- Un filtro, una compuerta.
- Dos filtros, una compuerta.
- Múltiples compuertas.

Tipos de *firewalls* en uso común actualmente

- *Firewalls* de paquetes.
- *Firewalls* basados en proxies.
- *Firewalls* de reescritura de paquetes.
- Pantallas.

Cuidado con los *firewalls*

- Están "de moda".
- Son solo una herramienta.
- **No** son una solución total.
- Hay que utilizarlos con buen criterio, y como parte de planes bien elaborados.

Disponibilidad

- **Objetivos:**
  1. Mantener el sistema funcionando todo el tiempo.
  2. Hacer que los recursos sean accesibles en el momento en que los usuarios lo necesiten.
- **Mecanismos:**
  1. Respaldos.
  2. Espejos.
  3. Redundancia física.

¿Por qué hacer respaldos?

Los datos son el componente más valioso de un sistema de cómputo.

> *"Para mi, los datos de los usuarios son de importancia inigualada. Cualquier otra cosa es generalmente reemplazable. Se pueden comprar más discos, más computadoras, más energía eléctrica. Pero si se pierden los datos, por un incidente de seguridad o por cualquier otra cosa, se van para siempre."*
>
> –Russell Brand

¿Por qué hacer respaldos?

- Errores de usuarios.
- Errores de administradores.
- Fallas de hardware.
- Fallas de software.
- Intrusos.
- Robo del equipo.
- Desastres naturales.
- Información histórica.

¿Qué se debe respaldar?

- Archivos de usuarios.
- Bases de datos del sistema.
- Directorios del sistema que sean especialmente importantes o que hayan sido modificados.

Tipos de respaldos

- Respaldo de *día cero*.
- Respaldos completos.
- Respaldos incrementales.

Estrategias de respaldo

Normalmente los respaldos completos e incrementales se utilizan conjuntamente para llevar a cabo la política de respaldo. Por ejemplo:

- Hacer un respaldo completo el primer día de cada mes.
- Hacer un respaldo incremental todas las noches, de todo lo que ha cambiado desde el principio de mes.

Recomendaciones para respaldos

- Determinar la periodicidad de respaldo de acuerdo a los datos que se estén respaldando.
- Respaldar periódicamente los directorios del sistema también.
- Hacer de vez en cuando un respaldo de todos los archivos y directorios en el disco.
- Usar un conjunto "rotatorio" de cintas.
- No usar las cintas para más de 100 respaldos.
- Hacer pruebas periódicas de recuperación de los respaldos.
- Llevar un índice de qué archivos están en qué cintas.

¿Cuánto tiempo conservar las cintas?

Depende de la información que esté almacenada en ellas. Algunas deben ser conservadas una semana, otras un mes, otras varios años, y algunas quizá para siempre.

"La cinta es barata, y ***rm*** *es para siempre..."*

Seguridad física de los respaldos

- Guardar los respaldos bajo llave.
- Quitar las cintas del drive después de hacer un respaldo.
- No guardar los respaldos en el mismo sitio que la computadora.
- Tener precauciones durante el transporte y manejo de los respaldos.

Espejos

Un espejo es un sistema que es una copia idéntica de otro.
Los espejos se suelen utilizar para tener la capacidad de proporcionar un servicio ininterrumpido a los usuarios. Si la máquina original falla, se activa el espejo y nadie nota el cambio (teóricamente).
El espejo no tiene que ser una copia exacta de todo el sistema, solo de los servicios que se quiere proteger.

Algunos ejemplos de espejos comunes

- Servidor de FTP.
- Servidor de WWW.
- Servidor de listas de correo.
- Servidor de correo electrónico.

Algunas consideraciones sobre los espejos

- ¿Cómo hacer la copia?
- ¿Con qué frecuencia hacer la copia?
- ¿Cómo verificar que no se copien cosas ya modificadas o perdidas?
- ¿Cómo verificar la integridad de la copia?
- ¿Cómo activar el espejo cuando el original falle?

Redundancia física

En un sistema de cómputo, es posible poner dos (o más) de cada elemento crítico, de manera que si el original falla, se pueda seguir usando la máquina.
Es como un "espejo de hardware".

Elementos que se pueden hacer redundantes

- Discos.
- Fuentes de poder.
- Tarjetas de red.
- Procesadores.
- Memoria.

Privacidad

- **Objetivos:**
  1. Mantener **mi** información privada.
  2. Proteger información tanto dentro como fuera del sistema (en la red).
- **Mecanismos:**
  1. Control de acceso (ya visto).
  2. Cifrado (se verá después).

Integridad, autenticidad y consistencia

- **Objetivos:**
  1. Verificar que los datos no sean modificados sin autorización.
  2. Verificar que los datos hayan sido realmente generados por su autor y no por un impostor.
  3. Asegurar que los programas se comporten como deben.
- **Mecanismos:**
  1. Listas de verificación.
  2. Sistemas confiables.
  3. Cifrado (se verá posteriormente).

Verificación de integridad

Un intruso puede modificar cualquier archivo del disco, por alguna de las siguientes causas:

- Esconder evidencia de la intrusión.
- Crear "puertas traseras" para hacer más fácil el acceso posterior.
- Plantar "caballos de Troya".
- Deshabilitar la seguridad o la contabilidad del sistema.

Listas de verificación de integridad

Son listas de archivos importantes a los que se monitorea periódicamente para ver si han sufrido algún cambio.
Por ejemplo, los directorios de binarios del sistema operativo nunca deben modificarse bajo condiciones normales. Un cambio en cualquiera de ellos es señal de actividad extraña en el sistema.
Por ello, es importante asegurar la integridad de ciertos archivos y directorios.

Protección y verificación de la integridad

Existen varias formas de proteger y verificar la integridad de los archivos existentes en el disco:

1. Poner bien los permisos de los archivos.
2. Usar dispositivos de solo lectura.
3. Hacer copias de comparación.
4. Generación y comparación de "firmas".

Poner bien los permisos

Es la primera línea de protección de la integridad de los archivos importantes. De esta forma se elimina la posibilidad de que un usuario normal, de forma accidental, modifique un archivo importante.
Esto no puede impedir que alguien que ha obtenido privilegios especiales en el sistema haga las modificaciones que quiera.
COPS: `ftp://ftp.super.unam.mx/pub/security/tools/cops.tar.gz`

Utilización de dispositivos de solo lectura

De ser posible, hay que poner los archivos que nunca se modifiquen en discos de solo lectura. Esto hace imposible que dichos archivos sufran modificaciones, aunque se tengan privilegios especiales.

Dispositivos de solo lectura

- CD-ROM y similares (la escritura es físicamente imposible).
- Dispositivos normales con protección por hardware (la escritura es físicamente imposible).
- Dispositivos normales con protección por software (con los privilegios apropiados, la escritura es posible).

¿Qué usar? Lo mejor que haya disponible.

Hacer copias de comparación

Consiste en realizar copias de los archivos importantes, y comparar los archivos con las copias periódicamente.
Obviamente, las copias tienen que estar en un sitio en el que no puedan ser modificadas.
Las copias tienen que ser generadas a partir de los datos "limpios", o no sirve de nada.

Utilización de firmas electrónicas

Una *firma electrónica* es un elemento de información que identifica de manera única a un archivo y a su contenido, de manera que si el archivo cambia, la firma cambia y es posible detectar la modificación.
Lo que se hace normalmente es generar las firmas de todos los archivos a proteger, almacenarlas en un sitio seguro, y periódicamente generar nuevamente las firmas y compararlas con las que están almacenadas.

Tipos de firmas electrónicas

- Información acerca del archivos.
- Firmas algebráicas.
- Firmas criptográficas.

Información acerca del archivo

La información acerca de las características del archivo puede servir como una firma muy rudimentaria. Por ejemplo:

- Nombre del archivo.
- Tamaño.
- Fecha de última modificación.
- Dueños.
- Características internas de almacenamiento (como el i-nodo en Unix).

La desventaja es que es fácilmente falsificable.

Firmas algebráicas

Son algoritmos que generan uno o más números basados en el contenido de un archivo, pero utilizando un algoritmo algebráico, tal como un cálculo polinomial.
El más común es el CRC (*Cyclic Redundancy Check*), que fue desarrollado originalmente para detectar errores en transmisiones de datos. Este algoritmo está implementado, por ejemplo, en el comando **sum** de Unix.
La desventaja es que es fácilmente falsificable. Es fácil encontrar archivos distintos que produzcan el mismo código CRC.

Firmas criptográficas

Son algoritmos criptográficos de una sola vía, que obtienen un número a partir del contenido de un archivo. Estos algoritmos, a diferencia del CRC, son *criptográficamente fuertes*, y es prácticamente imposible encontrar dos archivos diferentes que generen el mismo número.
Ejemplos de estos algoritmos son **SNEFRU** y **MD** (*Message Digest*).
También se les llama *Funciones Hash.*

Aplicaciones

Existen muchos programas que permiten realizar verificación de integridad. Los más utilizados en Unix son:

- COPS (*Computer Oracle and Password System*): `ftp://ftp.super.unam.mx/pub/security/tools/cops.tar.gz`
- Tripwire: `ftp://ftp.super.unam.mx/pub/security/tools/tripwire.tar.gz`

Sistemas Confiables (*Trusted Systems*)

El término "confianza" se refiere a que se confía en que el sistema sea capaz de mantener e implementar correctamente la política de seguridad del sitio.
Esta confianza se puede basar en un proceso extenso de evaluación y certificación del sistema, que toma en cuenta tanto las características técnicas del sistema como el ambiente en el que deberá operar.
Esta evaluación se puede basar en alguna de las especificaciones al respecto. La más importante de ella es el **Libro Naranja** (*Orange Book*).

¿Qué es un sistema confiable?

De acuerdo al Libro Naranja:

> ...un sistema que utiliza medidas de integridad de hardware y de software suficientes para permitir su uso en el procesamiento simultáneo de un rango de información desde no clasificada hasta altamente secreta, para un conjunto diverso de usuarios, sin violar ningún privilegio de acceso.

Características de un sistema confiable

Algunas características importantes en un sistema confiable son:

- Manejo de seguridad multinivel (controles de acceso obligatorios).
- Capacidad de identificación y autenticación.
- Manejo de *rutas confiables* dentro de la arquitectura.
- Implementación de mecanismos de auditoría confiables.

Auditoría

- **Objetivos:**
  1. Poder determinar quién hizo qué en el sistema.
  2. Proporcionar información para prevenir y corregir problemas.
- **Mecanismos:**
  1. Mantenimiento de bitácoras.
  2. Configuraciones correctas.

¿Qué es una bitácora?

Es un registro de actividad en el sistema.
Normalmente existen diferentes bitácoras en un sistema de cómputo, cada una generada por partes diferentes del mismo.

¿Por qué utilizar bitácoras?

Proporcionan una historia del pasado del sistema, que puede permitir detectar cosas que están saliendo mal, prevenir problemas futuros y averiguar qué ocurrió en problemas pasados.
Las bitácoras pueden servir para reconstruir un sistema, conducir una investigación, o incluso obtener un mejor servicio de mantenimiento por parte de un proveedor.

¿Qué registrar en una bitácora?

Algunas cosas que pueden resultar útiles son:

- Sesiones establecidas.
- Intentos fallidos de establecer sesiones.
- Terminación de sesiones.
- Accesos remotos.
- Aperturas, cerrados, borrados y renombrados de archivos.
- Cambios en privilegios de usuarios y procesos.

Comentarios sobre las bitácoras

- No sirven de nada si nadie las lee y analiza.
- La gran mayoría son generadas automáticamente por el sistema.
- Están sujetas también a modificaciones no autorizadas.
- Para mayor seguridad, mandarlas a otro sistema.

Configuraciones correctas

En muchas ocasiones, las bitácoras no se generan porque los servicios no están configurados para generarlas.
Hay servicios que no tienen la capacidad de generar bitácoras. Hay que descartarlos y obtener una versión que sí pueda hacerlo.

Incidentes y ataques

Comunmente se hace referencia a los problemas de seguridad con los términos *Incidente* y *Ataque*. Sin embargo, no significan lo mismo, y es importante conocer la diferencia. También es importante conocer los diferentes tipos de incidentes y ataques que existen, para poder reaccionar de manera apropiada.

¿Qué es un incidente?

Se le llama *incidente de seguridad* a cualquier evento que pone en riesgo cualquiera de los tipos de seguridad existentes en el sistema.
Un incidente puede ser desde un usuario que accidentalmente teclea `rm -rf *` hasta un ataque coordinado de manera internacional para robar datos de la máquina.

¿Qué es un ataque?

Se conoce como *ataque* un incidente ocasionado de manera intencional con el objetivo de causar un daño, obtener información, o hacer uso de recursos de manera no autorizada.
También se suele identificar con este término a cualquier intento de realizar dichas actividades ilícitas.

Tipos de incidentes

- Intencionales.
- No intencionales.
- Físicos y naturales.

Nótese la similitud con los tipos de amenazas.

Tipos de ataques

- Por su origen:
  - Internos.
  - Externos.
- Por su objetivo:
  - Al sistema.
  - En transmisión de datos.
  - En emisión de señales.
  - Físicos.

Ataques al sistema

Son todos los ataques que tienen como objetivo hacer uso de los recursos del sistema de manera no autorizada. Por ejemplo, obtener sesiones interactivas, modificar archivos, transferir archivos, crear cuentas, etc.

Ataques en transmisión de datos

Son todos los ataques que se pueden realizar sobre los datos mientras están en tránsito a través de una red. Incluyen:

- Intercepción.
- Modificación.
- Interrupción.
- Falsificación.

Ataques en emisión de datos

Implica la captura e interpretación de las señales electromagnéticas emitidas por los sistemas de cómputo, con el objetivo de lograr la reconstrucción de los datos que están siendo procesados.
El estandar TEMPEST guía el diseño, construcción e implementación de procedimientos y tecnologías que impidan este tipo de ataques.
!Suena a ciencia ficción, pero no lo es!

Ataques físicos

Comprenden la destrucción o descompostura intencional ocasionada en equipos de cómputo y la infraestructura relacionada (suministros eléctricos, cableado de redes, etc.)

¿Qué es el cifrado de datos?

Se conoce como *cifrar* (no encriptar) a la operación de transformar una serie de datos en otra (normalmente ininteligible), y a partir de la cual no se puede obtener, de manera obvia, la información original.
A la información original se le llama *texto en claro*, y al resultado de cifrarla se le llama *texto cifrado*.
A la operación inversa se le llama *descifrar*.

Llaves de cifrado

Las operaciones de cifrado y descifrado por lo regular funcionan con base en una *llave*, sin la cual la operación no puede llevarse a cabo de manera correcta.

¿Para qué sirve?

El cifrado de datos permite obtener:

- Privacidad.
- Integridad.
- Autenticidad.

Privacidad

Es asegurar que la información solo será utilizada por las personas autorizadas a ello. Mediante un algoritmo criptográfico se cifra la información original. Esta información entonces puede ser transmitida por canales públicos no seguros, pues aunque sea interceptada, no podrá ser utilizada a menos que se conozca la forma de descifrarla.

Integridad

Consiste en asegurar que la información no ha sido modificada de manera no autorizada. Mediante algoritmos criptográficos se generan "firmas" electrónicas de los mensajes, que a su vez pueden ser cifradas, y que de alguna manera son utilizadas, al recibir el mensaje, para asegurar que éste no ha sido modificado.

Autenticación

Es asegurar que la información fue generada por la persona correcta, y no se trata de una falsificación.
Mediante la utilización de firmas electrónicas asociadas a la persona que envía el mensaje, puede comprobarse que el mensaje fue enviado por la persona autorizada, y que no ha sido modificado o falsificado.

Tipos de sistemas de cifrado

De llave privada: Son los que utilizan la misma llave para cifrar y descifrar el mensaje.

De llave pública: Son los que utilizan una llave para cifrar el mensaje y otra para descifrarlo.

Sistemas de llave privada

La misma llave se utiliza para cifrar y para descifrar el mensaje. Por ejemplo: *Data Encryption Standard* (DES).
Ventaja: es muy rápido.
Desventaja: se necesita un canal seguro para el intercambio de la llave.

Fortaleza de DES

Durante mucho tiempo DES fue considerado prácticamente "irrompible". Actualmente, sin embargo, está comenzando a cuestionarse su fortaleza, sobre todo debido a que es posible construir *chips* que implementen el algoritmo en hardware, y que pueden probar millones de llaves por segundo, para realizar un ataque de *fuerza bruta*.

Sistemas de llave pública

Se utiliza una llave para cifrar el mensaje y otra diferente para descifrarlo. Por ejemplo: *Rivest-Shamir-Adleman* (RSA).
Una de las llaves, llamada *llave pública*, puede ser divulgada públicamente sin comprometer la seguridad del algoritmo.
La otra llave, llamada *llave privada*, debe ser de conocimiento únicamente para el dueño de la llave.
La llave pública puede descifrar mensajes cifrados con la llave privada, y viceversa.
Inventado por Diffie y Hellman en 1976.

Sistemas de llave pública

Ventaja: no se necesita un canal seguro para el intercambio de llaves.
Desventaja: puede ser muy costoso en términos de poder de cómputo.

Fortaleza de RSA

Se basa en la dificultad de factorizar un número muy grande (más de 200 dígitos), que es la base del algoritmo.
Es posible variar la longitud de la llave, con lo cual se puede incrementar su seguridad, aunque también se hace más lento.
Se estima que para romper una llave de 1024 bits por "fuerza bruta", se necesitarían 280,000 años de cómputo a 10,000 BIPS (Billones de Instrucciones por Segundo).

Otra aplicación de la criptografía

Los algoritmos *Message Digest*, también conocidos como funciones *hash*, son algoritmos criptográficos que no necesitan una llave.
Lo que hacen es que, dada una entrada, proporcionan una salida de longitud fija, y cuyo valor depende directamente de los datos de entrada.
Un buen algoritmo MD debe hacer muy difícil tener dos entradas diferentes que produzcan salidas iguales.
Ejemplos: MD2, MD4, MD5, Snefru, Haval, SHA.

MD5: Message Digest 5

Es el algoritmo de MD más seguro conocido hasta la fecha. Genera un resultado de 128 bits.
Aunque teóricamente pueden existir muchos mensajes que generen la misma "firma" con MD5, hasta la fecha no se ha encontrado ninguno.
El número de posibles firmas de MD5 es:
$2^{128} = 340,282,366,920,938,463,463,374,607,431,768,211,456$

¿Cómo se obtiene privacidad?

Esta es la aplicación obvia de la criptografía. Sin la llave correcta, es imposible saber qué dice un mensaje cifrado, y por lo tanto dicho mensaje se mantiene privado.

¿Cómo se obtiene autenticidad?

Utilizando algoritmos de llave pública:

1. El remitente cifra el mensaje utilizando su llave privada.
2. El receptor lo descifra con la llave pública del remitente.
3. Si el mensaje se descifra correctamente, solo puede haber sido cifrado (y por lo tanto generado) por el remitente original.

¿Cómo se obtiene integridad?

Una vez más, utilizando algoritmos de llave pública:

1. Antes de enviar el mensaje, el remitente obtiene una *firma* de él, utilizando un algoritmo de *Message Digest*.
2. La firma obtenida es cifrada con la llave privada del remitente.
3. El receptor descifra la firma con la llave pública del remitente.
4. El receptor obtiene por su cuenta la firma del mensaje.
5. Si la firma obtenida y la que fue enviada coinciden, el mensaje no ha sido modificado.

Fortaleza criptográfica

Es una medida de la seguridad del mensaje cifrado. Depende de:

- Qué tan secreta sea la llave.
- La dificultad de probar todas las llaves posibles.
- La dificultad de invertir el algoritmo sin conocer la llave.
- La existencia o no de "puertas traseras".
- La facilidad de descifrar un mensaje si se conoce una parte de él en su forma original.
- Las propiedades del mensaje original y si el atacante las conoce o no.

¿Seguridad por obscuridad? !No!

No es bueno basar la seguridad de un algoritmo criptográfico en el hecho de que nadie conozca el algoritmo.
Los mejores algoritmos conocidos a la fecha son públicos, y hay muchos que no han podido ser "rotos".

Ataques al cifrado

- De fuerza bruta.
- Texto en claro conocido.
- Texto en claro elegido.
- Criptoanálisis diferencial.
- Errores de implementación.

Ataques de fuerza bruta

Consisten en probar todas las posibles llaves, hasta dar con la correcta.
Aquí, la longitud de la llave y los requerimientos de cómputo del algoritmo determinan su fortaleza.
Este tipo de ataques son muy ineficientes. Por ejemplo, para una llave de 128 bits, si se pudieran probar un trillón de llaves por segundo, se tardaría más de 10^{13} años en probar todas las posibles llaves.

Texto en claro conocido

Si se tiene un bloque de texto en claro y el bloque correspondiente de texto cifrado (por ejemplo, un encabezado constante en los mensajes), se puede intentar obtener la llave, que entonces se utilizaría para descifrar el resto del mensaje.

Texto en claro elegido

Si el atacante puede hacer que la "víctima" cifre bloques de datos elegidos por él, la tarea del análisis y la obtención de la llave puede facilitarse. Una vez más, el objetivo es obtener la llave, para descifrar otros mensajes.

Criptoanálisis diferencial

Es un tipo de ataque por texto en claro elegido en el que el atacante logra el cifrado de muchos textos que difieren ligeramente entre sí, con el objetivo de comparar los resultados y obtener la llave.

Errores de implementación

Este es uno de los tipos más comunes de ataques exitosos. Un algoritmo puede ser perfecto, pero su implementación puede tener muchas fallas. Por ejemplo:

- Números "aleatorios" predecibles.
- Liberación accidental de las llaves.

Estándares en Criptografía

- DES es el estándar en algoritmos de llave privada.
- Algunos otros: Triple-DES, RC2, RC4, IDEA, Skipjack.
- RSA es el estándar en algoritmos de llave pública.
- Algunos otros: Diffie-Hellman, Markle-Hellman (roto).

Legislaciones sobre criptografía

Durante mucho tiempo, la criptografía fue terreno exclusivo de los gobiernos y las grandes corporaciones.
Actualmente, cuando todo mundo puede utilizarla, los gobiernos siguen queriendo tener el control.
En muchos países la utilización de criptografía sigue siendo ilegal, o teniendo muchas restricciones.

Algunas legislaciones sobre criptografía

Estados Unidos: La exportación de criptografía con llaves de más de 40 bits es ilegal.

Francia: La utilización de criptografía es ilegal.

Suiza: La utilización de criptografía es obligatoria en ciertas aplicaciones (por ejemplo, almacenamiento de registros médicos).

Aplicaciones

- El comando **crypt** de Unix.
- PGP (*Pretty Good Privacy*).
- S/Key.
- Tripwire.
- Kerberos.
- SecureRPC.

El comando crypt

Utiliza una variante del algoritmo utilizado por la máquina *Enigma*, que fue usada durante la Segunda Guerra Mundial por los alemanes.
El algoritmo utilizado por **crypt** es sumamente fácil de romper, e incluso existe un programa llamado *Crypt Breaker's Workbench*, que descifra automáticamente los mensajes cifrados con **crypt**.

PGP

Es un sistema de dominio público que utiliza una combinación de sistemas de llave pública (IDEA) y privada (RSA) para proporcionar privacidad y autenticación de mensajes.
PGP es el programa de cifrado más utilizado a nivel mundial actualmente. Su historia ha sido problemática y controversial, pero actualmente es utilizable de manera legal en todo el mundo.
`ftp://ftp.super.unam.mx/pub/security/tools/PGP/`

S/Key

Es un sistema que utiliza un algoritmo de *Message Digest* para evitar la transferencia de passwords a través de la red.
Permite eliminar el principal problema asociado a la utilización de passwords a través de la red.

Tripwire

Es un sistema que utiliza varios algoritmos de tipo *Message Digest* para proporcionar verificación de integridad de ciertos archivos de un sistema Unix.

Kerberos y Secure RPC

Son dos sistemas que utilizan combinaciones de algoritmos de llave pública y privada para lograr el establecimiento de sesiones cifradas entre dos máquinas Unix.

¿Qué son las políticas de seguridad?

Son códigos de conducta para la utilización de los sistemas de cómputo.
Especifican qué actividades no son permitidas, los pasos a seguir para lograr la protección adecuada, los pasos a seguir en caso de un incidente de seguridad, establece responsabilidades y derechos, etc.

¿Por qué utilizarlas?

- Sin políticas, no se tiene una infraestrucutra de seguridad general.
- Definen lo que es y lo que no es permitido.
- Ayudan a definir las necesidades de herramientas y procedimientos.
- Pueden ayudar a perseguir intrusos.
- Ayudan a ser "buenos vecinos" en la red.

La política de las políticas

- La elaboración e implementación de políticas suele ser el paso más difícil en la infraestructura de seguridad.
- Las políticas afectan a todos.
  - La gente se resiste a medidas que impidan la productividad.
  - Algunos simplemente se resisten al cambio.
  - Algunos se resisten a "la autoridad".
  - A algunas personas sencillamente les gusta molestar.
- Es imposible que todos estén de acuerdo. Hay que tratar de lograr una situación aceptable a la mayor parte.

Algunos puntos clave

- Antes de comenzar, hay que decidir la actitud general:
  - Lo que no esté permitido está prohibido.
  - Lo que no esté prohibido está permitido.
- Las nuevas políticas se tienen que conformar a las reglas y políticas que ya existan en la organización.
- Las políticas deben:
  - Ser implementables y vigilables.
  - Ser concisas y fáciles de entender.
  - Equilibrar protección con productividad.
  - Ser actualizadas periódicamente.

Algunos puntos clave

- El diseño de políticas debe ser un esfuerzo conjunto entre gente técnica, administrativa y directiva.
- Designar alguien que sirva cono el intérprete oficial de las políticas.
- Involucrar a una o más personas de cada grupo de la organización.
- Quienes tengan responsabilidades en la política también deben tener la autoridad para poder llevarlas a cabo.

Más puntos clave

- Todos los afectados por las políticas deben tener oportunidad de revisarlas antes de que se vuelvan oficiales.
- Que todo mundo sepa sus responsabilidades individuales. E.g.:
  - Los usuarios son responsables de los passwords y de reportar actividad sospechosa.
  - Los administradores son responsables de mantener la integridad de los sistemas.

Y más...

- La implementación exitosa requiere de un monitoreo continuo.
  - Planes de entrenamiento.
  - Explicar por qué son necesarias.
  - Explicar lo que se espera de la gente.
  - Estar al pendiente del cambio en la organización.

Elementos básicos de una política

- Definir acciones a tomar si se viola la política.
- Definir acciones a tomar si un usuario local viola una política en otro sitio.
- Definir contactos y responsabilidades hacia organizaciones externas.
- Hacerlas disponibles a todas las personas afectadas por ellas.

Política de seguridad del sitio

- Sirve como el plan de seguridad general.
- Define quién puede usar los recursos.
- Define el uso apropiado de los recursos.
- Define quien puede otorgar acceso a los recursos.
- Define las responsabilidades y derechos de usuarios y administradores.
- Define quién puede tener acceso de **root**.
- Define qué hacer con información delicada.

Políticas de uso apropiado

- Qué actividades son permitidas y cuales no.
- Actividades lucrativas.
- Uso personal y recreativo.
- Experimentación con la seguridad del sistema.
- Comportamiento de los superusuarios.
- Sanciones.
- Los usuarios deben leerla y firmarla antes de tener una cuenta.

Políticas de cuentas de usuarios

- Requerimientos para solicitar y mantener acceso a los recursos.
- Quiénes son los usuarios autorizados de cada sistema.
- Quién está autorizado a proporcionar el acceso.
- Mecanismos para creación y eliminación de cuentas.
- Condiciones para desactivación de cuentas.
- Reglas acerca de compartir cuentas.

Políticas de passwords

- Recomendaciones para elegir passwords seguros.
- Establecer claramente el carácter secreto y personal de los passwords.
- Recomendaciones de manejo de los passwords.
- Instrucciones para cambio de passwords.
- Políticas de envejecimiento de passwords.
- Sanciones.

Derechos y responsabilidades de los administradores

- Respetar la privacidad de los usuarios.
- Derecho de examinar los archivos de los usuarios si la situación lo amerita.
- Derecho de monitorear a los usuarios si la situación lo amerita.
- Todo tiene que estar por escrito.

Licencias y Derechos de autor

- Explicar qué son las licencias y para qué sirven.
- Qué productos tienen licencias.
- Qué implica la licencia.
- Sanciones por no respetar las licencias.

Operación en Internet

- Qué servicios se pueden usar.
- Responsabilidades de seguridad del usuario.
- Responsabilidades del administrador.
- Reglas de "etiqueta".

¿Qué es un oficial de seguridad?

Es una persona encargada de monitorear y mantener la seguridad en un sistema de cómputo.
Tradicionalmente esta actividad es realizada por el mismo administrador u operador.
Sin embargo, cada vez más se está convirtiendo en una actividad de tiempo completo.

Características ideales de un OSS

- Conocimientos técnicos.
- Experiencia en el manejo de los sistemas.
- Familiarización con la organización, su infraestructura y su forma de operación.
- Capacidad de trato humano.
- Capacidad de trabajo bajo presión.
- Alto sentido de la responsabilidad.
- Integridad y confiabilidad a toda prueba.

¿Qué obligaciones tiene?

(no necesariamente todas, depende de las políticas de la organización)

- Monitorear continuamente los sistemas a su cargo.
- Investigar y corregir incidentes de cualquier tipo.
- Comunicación y coordinación con otras organizaciones.
- Educar a los usuarios.
- Difundir información.

¿De qué privilegios goza?

(una vez más, depende de las políticas del sitio)

- Acceso a información privilegiada o confidencial relativa a los sistemas.
- Acceso a información referente a incidentes de seguridad.
- Privilegios de administración en los sistemas.
- Capacidad de emitir recomendaciones (o incluso órdenes) a los administradores.
- Permiso para experimentar con la seguridad de los sistemas, con el objetivo de descubrir nuevos problemas.

¿Cuáles son sus limitaciones?

(dependiendo de las políticas...)

- No monitorear sin razón a los usuarios.
- No violar la privacidad de los usuarios.
- No impedir la productividad de los usuarios.
- No dejar en mal funcionamiento el sistema.
- No difundir información sensible o confidencial.
- "No hacer cosas buenas que parezcan malas" (y tampoco cosas malas).

Consideraciones éticas

- Los usuarios tienen derecho a su privacidad.
- Los usuarios tienen derecho a su libertad de acción, dentro de lo establecido por la organización.
- Los usuarios son seres humanos, no pueden (ni van a) obedecer órdenes o recomendaciones irracionales.
- No abusar de los privilegios.

¿Cómo se establece la confiabilidad?

Es muy difícil decir a primera vista si una persona es confiable o no. Sin embargo, hay algunos factores curriculares que pueden ayudar:

- Recomendaciones.
- Desempeño en trabajos pasados.
- Historia criminal.
- Comportamiento en otras actividades no relacionadas con seguridad.

Formas de administración de la seguridad

Centralizada: Concentrar todos los privilegios y responsabilidades en un solo sitio.

Distribuida: Cada quien es responsable de lo suyo. Puede haber una autoridad central que otorgue asesorías y emita recomendaciones.

Regla #1: !No Asustarse!

- ¿Realmente es una violación de la seguridad?
- ¿Se ha hecho algún daño?
- ¿Es importante obtener y proteger la evidencia que pueda servir en una investigación?
- ¿Es importante tener el sistema en funcionamiento normal lo más rápido posible?
- ¿Cómo asegurar que no haya archivos modificados?
- ¿Importa si alguien se entera del incidente?
- ¿Puede suceder nuevamente?

Regla #2: !Documentar TODO!

Es muy importante registrar todo lo que sucede, **detalladamente**. Se pueden usar las siguientes técnicas:

- Escribir todo lo que se hace y se encuentra en una libreta.
- Imprimir los archivos de texto que se examinen.
- Utilizar el comando **script** (en Unix) para registrar las sesiones de trabajo.

Todo lo que esté impreso en papel debe tener la fecha y la firma de la persona responsable.
Es importante mantener copias de todo en papel.

Detección

- Descubrir al intruso en el acto.
- Deducirlo con base en cambios en el sistema: cuentas nuevas, archivos nuevos, modificación de archivos de datos, discrepancias en la contabilidad, etc.
- Deducirlo con base en el comportamiento del sistema: caídas frecuentes, mal desempeño, negación de servicio, etc.
- Recibir un mensaje del administrador de otro sitio reportando actividad extraña originada desde la máquina local.

Cómo detectar a un intruso en el acto

- Un usuario conectado al mismo tiempo desde lugares diferentes.
- Un usuario ejecutando un programa que no suele usar.
- Un usuario haciendo uso excesivo de la red.
- Un usuario que no tiene un modem conectado a través de la línea telefónica.
- Un usuario ejecutando comandos como superusuario.
- Un usuario conectado en horas inusuales.

¿Qué hacer?

- Ignorarlo.
- Tratar de comunicarse con él.
- Tratar de rastrear la conexión.
- Cortar su conexión.

Comunicarse con el intruso

Es importante registrar todo lo que diga o envie (si lo hace).
A veces basta hacer el intento de comunicación para que el intruso desaparezca.
A veces, es posible hacer entender al intruso el daño que está causando.

Rastrear la conexión

El comando **who** (en Unix) despliega desde dónde está conectado cada usuario.
Con el comando **finger** es posible ver qué usuarios están en la máquina desde donde el intruso está conectado.
Es importante comunicarse con el administrador del sitio remoto.

Cortar la conexión

- Apagar la computadora.
- Matar los procesos del intruso.
- Dar de baja la máquina.

Revisión de las bitácoras (auditoría)

Una cuidadosa revisión puede decirnos exactamente qué pasó. Al revisarlos, hay que buscar por cosas fuera de lo común:

- Usuarios en sesión en horas extrañas.
- Intentos fallidos de conexión repetidos.
- Uso no autorizado de privilegios administrativos.
- Usuarios conectándose desde sitios no conocidos.

Contención y reparación de daños

- Cuentas nuevas.
- Cambio de password de cuentas existentes.
- Cambio de permisos en ciertos archivos.
- Archivos nuevos con privilegios especiales.
- Programas del sistema modificados o reemplazados.
- Modificaciones al sistema de correo electrónico.
- Modificaciones al sistema de acceso por modem.

Notificación

- Al personal de administración del sistema.
- Al personal directivo.
- Al personal usuario.
- A las autoridades correspondientes.
- A organismos de seguridad.
- A la prensa.

Investigación

- ¿Vale la pena? ¿Se obtiene algún beneficio?
- ¿Se tienen los recursos (materiales, legales, humanos) para hacerlo?
- ¿Ayuda a la restitución de los daños?
- ¿Basta con cortar el acceso a los intrusos, o podemos ir más allá?

Resumen: pasos a tomar

1. Identificar y entender el problema.
2. Detener el daño.
3. Confirmar el diagnóstico y determinar el daño.
4. Recuperar el sistema.
5. Lidiar con la causa del problema.
6. Hacer tareas de recuperación relacionadas.

Algunos recursos útiles

- Herramientas de monitoreo y control.
- Respaldos.
- Redundancia y espejos.
- Recursos fuera de sitio.

¿A quién acudir?

A veces se necesita una asesoría externa. Hay organismos que están dedicados a proporcionar ayuda y asesoría a sitios que tengan problemas de seguridad.
Es importante tener en cuenta que cada organismo tiene una comunidad objetivo bien definida, y puede ser difícil que otorguen ayuda a gente fuera de esa comunidad.
También hay que estar conscientes de que estos organismos no van a hacer todo por nosotros, normalmente se limitan a ayudar en el análisis y a emitir recomendaciones de acciones a tomar.

Organismos de Seguridad

MxCERT: Todo México. `http://www.mxcert.org.mx/`

Area de Seguridad en Cómputo, UNAM: UNAM.
`http://www.super.unam.mx/asc/`

CERT (Computer Emergency Response Team): Todo el mundo.
`http://www.cert.org/`

!GRACIAS!