Introducción a la Ciberseguridad y sus aplicaciones en México

Rocío Aldeco Pérez
Gina Gallegos García
Lil María Rodríguez Henríquez
(Editoras)

ACADEMIA MEXICANA DE COMPUTACIÓN, A, C.

Introducción a la Ciberseguridad y sus aplicaciones en México

Autores:
Rocío Aldeco Pérez
Gualberto Aguilar Torres
Nareli Cruz Cortés
Luis J. Domínguez Perez
Ponciano J. Escamilla Ambrosio
Gina Gallegos García
Miguel A. León Chavez
Juan C. López-Pimentel
Raúl Monroy Borja
Lil M. Rodríguez Henríquez
Francisco J. Rodríguez Henríquez
Abraham Rodríguez Mota
Moisés Salinas Rosales
Alejandra G. Silva Trujillo

Coordinadora: Gina Gallegos García
Colaboradores capítulo 2: Alfonso F. De Abiega L'Eglisse y Kevin A. Delgado Vargas
Colaborador capítulo 3: Victor G. Reyes Macedo
Colaboradores capítulo 5: Bruno Ramos Cruz, Juan J. Gomez Olvera, y Saúl Pomares Hernández

Primera edición: 2020
Academia Mexicana de Computación, A. C.

ISBN: 978-607-98941-2-2

Corrección de estilo: Rocío Aldeco Pérez y Lil M. Rodríguez Henríquez.
Diseño de portada: Mario A. Vélez Sánchez.
Cuidado de la edición: Luis E. Sucar Succar.

Este libro se realizó con el apoyo del CONACyT, Proyecto 311180.

Introducción a la Ciberseguridad y sus aplicaciones en México.

Autores

Rocío Aldeco Pérez
Gualberto Aguilar Torres
Nareli Cruz Cortés
Luis J. Domínguez Perez
Ponciano J. Escamilla Ambrosio
Gina Gallegos García
Miguel A. León Chavez
Juan C. López-Pimentel
Raúl Monroy Borja
Lil M. Rodríguez Henríquez
Francisco J. Rodríguez Henríquez
Abraham Rodríguez Mota
Moisés Salinas Rosales
Alejandra G. Silva Trujillo

Agradecimientos

Agradecemos a la Academia Mexicana de Computación, en especial a los integrantes que colaboraron con el desarrollo del libro: Rocío Aldeco Pérez, Gualberto Aguilar Torres, Nareli Cruz Cortés, Luis J. Domínguez Perez, Ponciano J. Escamilla Ambrosio, Gina Gallegos García, Miguel A. León Chavez, Raúl Monroy Borja, Lil M. Rodríguez Henríquez, Francisco J. Rodríguez Henríquez, Abraham Rodríguez Mota, Moisés Salinas Rosales, y Alejandra G. Silva Trujillo, por su valiosa colaboración y tiempo dedicado a este libro. Agradecemos particularmente a Gina Gallegos García por su coordinación en este proceso, también a Rocío Aldeco Pérez y a Lil M. Rodríguez Henríquez por la correción del estilo. Especialmente agradecemos a Luis E. Sucar por sus valiosos comentarios, sugerencias y el impulso encaminado hacia la publicación de este libro.

Agradecemos al CONACyT por su financiamiento para la creación de esta obra.

Prólogo

¡Bienvenido!
En tus manos tienes una obra que es resultado de la colaboración de Profesores-Investigadores de Universidades Públicas, Privadas y Centros de Investigación en México.

Cada uno de ellos preocupados y ocupados en la seguridad de tu información, desde que se crea, almacena, procesa, transmite y se recibe desde un dispositivo electrónico a través de las redes de comunicación.

Proteger la información contra revelaciones no autorizadas (servicio de confidencialidad), contra modificaciones no autorizadas (servicio de integridad), contra la creación por terceras personas (servicio de autenticación), contra el acceso no autorizado (servicio de control de acceso) y contra la negación de alguno de sus procesamientos (servicio de no rechazo) son problemas que ha enfrentado la humanidad en el transcurso de su historia y que hoy en día se han agravado con el uso de las tecnologías de la información y la comunicación.

La ciberseguridad ha sido definida por el ITU-T X.1205 como el conjunto de técnicas que intentan proteger el entorno cibernético de un usuario o de una organización. Este entorno incluye a los seres humanos, redes de comunicación, dispositivos de comunicación, los sistemas de software, incluyendo los sistemas

operativos, el almacenamiento, procesamiento y transmisión de la información, así como los sistemas que están directa o indirectamente conectados a las redes.

Los servicios de seguridad se pueden implementar usando algoritmos y protocolos criptográficos especificados por diferentes organismos de estandarización internacionales y nacionales, algunos de los cuales se revisan en este libro, pero además sensibilizado a los usuarios, organizaciones, empresas y gobiernos de la importancia de la Ciberseguridad y este es uno de los objetivos de ese libro: sensibilizar, educar, orientar y difundir los esfuerzos individuales y colectivos de los autores.

Para ello hemos organizado el libro en 8 capítulos que revisan brevemente la Ciberseguridad y el Ciberespacio, las líneas de Investigación que se desarrollan en nuestro país, conceptos básicos de la criptografía moderna y de la criptografía pos-cuántica, aplicaciones de la criptografía en las en las bases de datos, redes vehiculares y el Internet de las Cosas, para finalmente proporcionar información a los interesados en estudiar un posgrado en Ciberseguridad. De esta forma la organización del libro es la siguiente:

El capítulo I titulado Ciberseguridad y ciberespacio presenta definiciones, requerimientos y normatividad internacional y nacional.

Los esfuerzos de la comunidad científica en México se describen brevemente en el capítulo II, titulado Ciberseguridad en México. Dichos esfuerzos se presentan también a través de una lista de oportunidades en dónde realizar estudios de posgrado en Ciberseguridad en el país, describiendo los programas de las instituciones nacionales.

El capítulo III, titulado Criptografía Moderna, presenta los conceptos básicos de la Criptografía de Llave Simétrica, de la Criptografía de Llave Asimétrica y de las Funciones Hash.

En el capítulo IV se discuten las firmas digitales y la necesidad de una Infraestructura pública para poder hacer uso de la criptografía asimétrica de forma segura.

Existen múltiples aplicaciones que requieren de algoritmos criptográficos para brindar servicios o productos de manera segura. En el Capítulo V, titulado aplicaciones selectas de la criptografía moderna, se discuten dos aplicaciones vanguardistas la seguridad en bases de datos como servicio y la seguridad en la redes vehiculares. La primera aplicación considera el escenario en qué el dueño de la información la delega a un proveedor que no es confiable, la segunda discute cómo se puede brindar seguridad en las comunicaciones dentro de las redes vehiculares ad-hoc.

Con el desarrollo de la tecnología hoy existen computadoras cuánticas, que aunque con capacidad de cómputo y almacenamiento limitado, en un futuro cercano serán capaces de resolver los problemas computacionales difíciles de manejar por las computadoras clásicas y en los que está basada la criptografía de llave asimétrica. De aquí la necesidad de diseñar y estandarizar nuevos algoritmos de llave asimétrica que sean resistentes a los ataques de computadoras post-cuánticas y clásicas. Tema que se desarrolla en el capítulo VI titulado Criptografía Pos-cuántica.

El capítulo VII discute los distintos enfoques para el diseño de protocolos de seguridad, la notación que se utiliza para especificar protocolos y en específico se decriben dos protocolos el NSPK y Wide Mouth Frog y como éstos pueden ser atacados.

Finalmente, en el Capítulo VIII se describe que es el Internet de las cosas (IoT), los dispositivos que intervienen y sus vulnerabilidades, y los retos para proveerlos de seguridad.

Los autores agradecen profundamente tu interés en la temática de "Introducción a la Ciberseguridad y sus Aplicaciones en México", esperando que su lectura sea comprensible, didáctica, ilustrativa y agradable, de tal manera que resuelva tus principales dudas en esta apasionante área de la informática y que te motive a incorporate a esta área de conocimiento en un futuro próximo.

Dr. Miguel Angel León Chávez

Dr. Francisco Rodríguez Henríquez

Índice general

Capítulo 1

Ciberseguridad y ciberespacio

Resumen

Escuchar frases en donde se utilizan palabras como ataques cibernéticos, ciberamenazas, *hackers*, virus informático, *software* malicioso, suplantación de identidad y ciberfrude por mencionar algunas, es cada vez más común.

Esto, hace referencia a los múltiples riesgos que existen al momento de conectar un dispositivo a la red formando un ecosistema mejor conocido como ciberespacio.

Este capítulo presenta un marco conceptual que hace referencia a los términos más utilizados en la actualidad, siendo estos: ciberseguridad y ciberespacio.

1.1. Origen de la ciberseguridad

Si eres relativamente nuevo en temas de ciberseguridad, seguramente no sabías que existe el día Internacional de la Ciberseguridad, el cual se celebra el día 30 de noviembre. Esta fecha, que se comenzó a celebrar hace casi 40 años, fue establecida por la Association for Computing Machinery [1], con el propósito principal de crear conciencia en la sociedad, de la seguridad en los sistemas interconectados. Sin embargo, todo indica que seguimos sin comprender la importancia que tiene la ciberseguridad en varios sectores de la población. Nos hemos preocupado únicamente por seguir mejorando las tecnologías de la información sin detenernos y analizar que la ciberseguridad es un tema que debería haber evolucionado con la misma velocidad.

Se está convirtiendo en una costumbre despertar cada día con al menos una noticia de un ciberataque a gran escala en el mundo y los informes anuales de las instituciones y organismos relacionados con temas de seguridad lo confirman, ya que la mayoría coincide en que año con año se han incrementado considerablemente los ciberataques. Ahora bien, debemos estar preparados para un incremento aún mayor en este año 2020 derivado principalmente de la pandemia que seguimos viviendo. La razón de este muy probable crecimiento es muy sencilla, se debe principalmente al incremento considerable de dispositivos conectados a internet [2] [3].

Seguramente te estarás preguntando por qué no se han podido combatir dichos ciberataques y por qué siguen en aumento. Las respuestas son varias y generan muchos debates. En primer lugar podemos argumentar que la velocidad con la que evolucionan las tecnologías de la información ocasiona que las vulnerabilidades se incrementen con el mismo ritmo. La proliferación de nuevos objetos conectados a internet aumenta el universo de trabajo para los cibercri-

minales. Si a esto le agregamos una falta de interés por parte de los directivos para proteger sus empresas, instituciones, trabajadores, clientes, etc., debido a que creen que no serán atacados por cibercriminales. Lo anterior es un exceso de confianza, puesto que parten de la suposición que los ciberataques están diseñados para grandes empresas, lo que deriva en el éxito de los ciberataques.

Ahora bien, podríamos pensar que con una tecnología correcta podemos estar seguros, la realidad es que no siempre sucede así, de hecho casi nunca, la principal razón es que muchos de estos ciberataques tienen éxito debido al factor humano. Las empresas cada vez gastan más en ciberseguridad, recurren a lo último en tecnología para tapar las brechas de seguridad, pero el eslabón más débil en ciberseguridad sigue siendo el hombre.

Y aunque la seguridad ha existido desde hace muchos años, ésta se caracterizó principalmente por concientizar en el buen uso de la información por parte de los empleados y así garantizar la seguridad de la organización. Sin embargo, con la acelerada evolución de la tecnología, surgieron nuevos riesgos que hicieron que este tipo de seguridad se volviera inoperante.

Hoy sabemos que el evento que impulsó el cambio en la seguridad de las empresas fue ocasionado por el primer virus informático en los años ochenta. Para la década de los noventa surgieron los primeros ataques a través de internet y a partir del año 2000 con el inicio de las redes sociales, los riesgos comienzan a incrementarse y observamos los primeros fraudes en internet. En los últimos años, con el surgimiento del Internet de las Cosas, podemos encontrar millones y millones de cosas conectadas a internet, desde dispositivos que tenemos en casa que nos facilitan muchas labores, hasta sistemas o servicios que ayudan a mantener con vida a personas [2] [3]. Por lo tanto, los riesgos han aumentado

considerablemente y esta es una de las razones por las que la Ciberseguridad es y seguirá siendo un elemento fundamental para la sociedad.

1.2. Necesidad de la ciberseguridad

Escuchar frases o palabras como ataques cibernéticos, ciberamenazas, *hackers*, virus informático, *malware*, *phishing*, ciberfraude, es cada día más común. Todo lo anterior, hace referencia a los múltiples riesgos que existen al momento de conectar un dispositivo electrónico a la red. Recordemos algunas frases que se han hecho muy populares en ciberseguridad:

- "El único sistema seguro es aquel que está apagado, desconectado, enterrado en un refugio de cemento, rodeado por gas venenoso y custodiado por guardias bien pagados y bien cerrado. Aun así yo no apostaría mi vida por él." Gene Spafford.

- "Un día seguro, ¡Seguro es un gran día!" Anónimo.

- "Las empresas invierten millones en *firewalls*, cifrado y dispositivos para acceder de forma segura, y es dinero malgastado, porque ninguna de estas medidas corrige el eslabón más débil de la cadena de seguridad: la gente que usa y administra los ordenadores." Kevin Mitnick.

- "Mi mensaje para las empresas que piensan que no han sido atacadas es: no estás buscando lo suficiente." James Snook.

- "Existen dos tipos de empresas: las que han sido hackeadas y las que aún no saben que fueron hackeadas." John Chambers.

Todas estas frases nos deben hacer reflexionar para tomar en serio a la ciberseguridad, si es que aún no lo hemos hecho. Es difícil imaginar algún sector de la sociedad que en la actualidad no necesite echar mano de la ciberseguridad. En los últimos años el sector financiero ha sido uno de los sectores más vulnerables a los ciberataques, ya que entre otras cosas incluye en sus bases datos información muy importante sobre sus clientes. También hemos visto como el sector de la salud ha sido constantemente atacado ya que al igual que el financiero, manejan información muy importante de personas. El sector educativo es uno más, por lo tanto, debemos ser conscientes de que no importa el sector o el tamaño de una empresa, todos los datos que poseen pueden ser objetivos de los ciberdelincuentes [4] [5].

Y es que el objetivo de los ciberdelitos es obtener información con la cual posteriormente los ciberdelincuentes pueden extorsionar, robar, amenazar, secuestrar, etc. En otras palabras, buscan información para posteriormente darle un uso que la mayoría de las veces no es correcto. Las formas para obtener información son múltiples por ejemplo, recurren a distintas técnicas como *phishing*, *vishing*, *spam*, virus, *malware*, ingeniería social, entre otras más.

Ahora bien, con lo mencionado hace un momento y con todas las noticias que escuchamos día con día sobre ciberataques, debemos ser conscientes de que la inversión en ciberseguridad es la única de la que no se debería prescindir en ningún momento. Aunque podemos ver que las empresas son cada vez más conscientes de que la ciberseguridad es esencial para el buen funcionamiento de su negocio, aún falta bastante camino para conseguir que ésta sea parte inherente de las empresas.

1.3. Ciberseguridad y ciberespacio

El ciberespacio es un entorno no físico, virtual, resultado de la interacción entre las personas y los equipos de cómputo conectados a una red. Es un término utilizado ampliamente en la actualidad y su objetivo es la transmisión de información.

La cantidad de información en el ciberespacio sigue creciendo y lo seguirá haciendo ya que cada vez seguimos conectando más y más dispositivos a la red. Esta transformación digital que se está viviendo, se puede entender fácilmente con el Internet de las Cosas (IoT). La digitalización y conexión de dispositivos desde los personales hasta los dispositivos del sector salud encargados de mantener con vida a pacientes, ha incrementado la cantidad de datos en el ciberespacio.

1.4. Principales amenazas en el ciberespacio

Sabemos que el año 2020 quedará marcado como el año donde el mundo se enfrentó a una de las más grandes pandemias en la historia y con esto, a un cambio en el modo de vida de las personas. Este cambio tuvo como elemento central el uso de las tecnologías de la información a nuestro alrededor. Si bien, el uso de nuevas tecnologías evolucionaba rápidamente, aún existían muchos sectores de la población que no las integraban por completo. Con la llegada de la pandemia, el mundo tuvo la necesidad de adaptarse a una nueva forma de interactuar, desde la forma de convivir con la familia, el estilo de trabajar, el modo de adquirir productos, la forma impartir y tomar clases, entre otras actividades. La pandemia que estaba iniciando ocasionó que el uso de herramientas tecnológicas se convirtiera en una necesidad primordial para seguir desarrollando las actividades diarias. Todo esto ocasionó que en pocos meses la cantidad de dispositivos

conectados a internet aumentara considerablemente y con ello también, la cantidad de amenazas y riesgos, ya que la sociedad no tuvo tiempo de capacitarse en materia de ciberseguridad.

Existe una lista interminable de amenazas en el ciberespacio [6], sin embargo, en esta sección mencionaremos solo aquellos que han marcado la transformación de la ciberseguridad.

1.4.1. Secuestro de datos

El secuestro de datos mejor conocido como *ransomware* se puede considerar uno de los principales ciberataques en los últimos años. Ha generado grandes ganancias para los ciberdelincuentes en todo el mundo y como muestra tenemos a *Wannacry* que se dio a conocer en el año 2017, ocasionando muchos problemas alrededor del mundo. Se puede considerar como un *software* malicioso que al entrar a nuestro equipo realiza un proceso de cifrado a los archivos. Posteriormente, se mostrará una ventana donde se solicita un pago de rescate, que en los últimos años consiste en un pago en criptomonedas. En otras palabras, el ransomware es un *malware* que cifra la información y para poder obtener la llave con la cual se podrían descifrar dicha información, es necesario realizar un pago. Sin embargo, es muy importante tener presente que el hecho de realizar el pago no garantiza que se pueda recuperar la información.

1.4.2. Suplantación de identidad

El también conocido como *phising* es una técnica normalmente utilizada por ciberdelincuentes para tratar de engañar a usuarios de ciertas páginas web, con el propósito de obtener información confidencial y que posteriormente podría ser utilizada para llevar a cabo un fraude. La información que puede ser

obtenida mediante esta técnica va desde los datos personas incluyendo las contraseñas, hasta datos de cuentas bancarias. Esta técnica consiste en suplantar la imagen de un sitio web, para hacer creer a la víctima que los datos solicitados son de un sitio oficial, cuando realmente no es así. Algunas variantes de esta técnica son mensajes de texto al teléfono móvil y llamadas telefónicas, ambas son técnicas que tienen el mismo propósito.

1.4.3. Robo de identidad

El robo de identidad es considerado también uno de los delitos que han incrementado en los últimos años. El objetivo principal de estas acciones en la mayoría de los casos es la comisión de fraudes. El robo de identidad ocurre cuando una persona utilizando distintas técnicas como *phishing*, ingeniería social, *vishing*, *spam* o alguna otra, adquiere información personal la víctima de forma no autorizada con el propósito de cometer algún delito como fraude o estafa usando la identidad de la víctima.

1.4.4. Ataques DoS

Los ataques DoS o ataques de denegación de servicio tienen como objetivo principal inhabilitar un sistema o servicio. De forma muy general, este ataque consiste en realizar un número masivo de peticiones a un mismo servicio en un instante de tiempo. Esto ocasiona en muchos casos que se consuma el total de los recursos originalmente destinados y cuando esto ocurre, el sistema o servicio empezará a rechazar ciertas peticiones ya que no tendrá la capacidad suficiente para seguir dando respuesta. Esto hará que los usuarios de este servicio sean incapaces de acceder a él.

1.5. Normatividad

En México existen pocas leyes en materia de seguridad informática. En esta sección mencionamos algunas de estas leyes. En particular, la técnica de firma digital en México es la más regulada a través de diversas leyes que describimos a continuación. El tema de firmas digitales lo veremos más detalladamente en el Capítulo §4

1.5.1. Ley Federal de Protección de Datos Personales en Posesión de Particulares

En el Artículo 16º de la Constitución Política de los Estados Unidos Mexicanos [27] dice en su segundo párrafo:[1]

"Toda persona tiene derecho a la protección de sus datos personales, al acceso, rectificación y cancelación de los mismos, así como a manifestar su oposición, en los términos que fije la ley, la cual establecerá los supuestos de excepción a los principios que rijan el tratamiento de datos, por razones de seguridad nacional, disposiciones de orden público, seguridad y salud públicas o para proteger los derechos de terceros."

Con la Ley Federal de Protección de Datos Personales en Posesión de Particulares [28] y su Reglamento [29], esencialmente las personas físicas o morales privadas que traten con información de caracter personal, deberán de cumplir una serie de obligaciones para proteger la información bajo su resguardo de las personas con que interactúa.

En particular, esta ley establece que no se pueden compartir datos con terceros ni cambiar el uso de la información personal originalmente estipulado,

[1] Párrafo adicionado DOF 01-06-2009

salvo previa autorización explícita del cliente. Para estas autorizaciones además, es posible el uso de la firma electrónica.

Asímismo, es obligación de los particulares avisar sobre fallas de seguridad que haya habido en la empresa y si estás fallas derivaron en alguna afectación sobre la información del cliente. En particular si se filtró información.

El Instituto Nacional de Transparencia, Acceso a la Información y Protección de Datos Personales y su Coordinación de Protección de Datos Personales se encargan de dar apoyo técnico para el cumplimiento de esta Ley.

1.5.2. Ley Federal de Protección de Datos Personales en Posesión de Sujetos Obligados

En el Artículo 6º de la Constitución Política de los Estados Unidos Mexicanos, en la sección A, Puntos II al IV, dice que:

"II La información que se refiere a la vida privada y los datos personales será protegida en los términos y con las excepciones que fijen las leyes. III. Toda persona, sin necesidad de acreditar interés alguno o justificar su utilización, tendrá acceso gratuito . . . a sus datos personales o a la rectificación de éstos."

que sumado al Artículo 16º de la misma constitución utilizado para la elaboración de la Ley Federal de Protección de Datos Personales en Posesión de Particulares, dan pie a la elaboración de esta ley. De esta manera, se garantiza el derecho a la privacidad previsto en diversos tratados internacionales en materia de derechos humanos, entre los que podemos destacar:

- Convención Americana de Derechos Humanos, Artículo 11º [30]
- Pacto Internacional de Derechos Civiles y Políticos, Artículo 17º [31]

Adicionalmente, la Suprema Corte de Justicia de la Nación declaró, al resolver la contradicción de tesis 293/2011, que dichas jurisprudencias son jurídicamente vinculantes para México. [32]

Esta ley es una ley marco, por lo que cada estado es responsable de establecer sus propias leyes estatales en este tema. Asimismo, el INAI será el encargado de darle seguimiento y apoyo técnico a las entidades y sujetos obligados para el cumplimiento de esta ley.

1.5.3. Ley de Firma Avanzada

El Servicio de Administración Tributaria (SAT) fue la primera instancia a nivel nacional en implementar y oficializar uso de la firma digital; sin embargo, diversas leyes fueron incluyendo este esquema de firma digital en diversos trámites de diversas dependencias de gobierno.

Para el caso de la seguridad de la información en posesión de los ciudadanos y de las empresas públicas y privadas, se creó la Ley Federal de Protección de Datos Personales en Posesión de los Particulares, mencionada anteriormente. Entre los cambios a esta ley, está como opción el uso de la firma electrónica para las diversas autorizaciones de cambios de uso, actualizaciones, bajas o transferencias a terceros de la información personal de las personas. De igual manera estas acciones estan descritas en la Ley Federal de Protección de Datos Personales en Posesión de Sujetos Obligados aplicable para las instituciones públicas.

En el caso de la Ley Federal de Transparencia y Acceso a la Información Pública Gubernamental [33], se requieren mecanismos para la verificación de la integridad de los datos. Esto se puede implementar mediante la firma digital; sin embargo, la ley no establece ningún mecanismo explícito.

Para las transacciones comerciales, el Código de Comercio [34], establece los requisitos para que las operaciones comerciales puedan hacerse de manera legal mediante medios electrónicos. Asimismo, establece ciertos lineamientos sobre lo que debe de llevar a cabo una firma electronica, y otros requisitos para los proveedores respecto a los certificados digitales de las firmas electrónicas y ciertas consideraciones para autoridades certificadoras del exterior del país.

En el caso del Código Fiscal de la Federación [35], utilizado principalmente por el SAT para la recolección de impuestos, establece directrices generales, entre las cuales están:

- Contenido de documentos digitales
- Sello digital de Hacienda
- Certificados de firmas digitales y sus características
- Emisores
- Procesos alrededor del pago de impuestos vía electrónica

La Ley de Firma Electrónica Avanzada [36], una de las últimas leyes en aprobarse durante el sexenio 2006-2012, cubre aspectos generales de aplicación de la firma digital. En particular, da certeza jurídica a la firma digital, cuando esta cumple con ciertos requisitos. También se cubren aspectos de los ceritificados digitales y sobre quién los emite.

A principios del sexenio 2012-2018 se emitió un Decreto de Austeridad [37] para entidades federales con el fin de reducir el uso del papel, y por ende, sobre la digitalización de documentos legales, utilizando la firma digital. Es en ese sexenio que el proceso de digitalización comenzó, teniendo sus mayores logros hacia

finales del mismo. En ese periodo se aceleró el uso de la firma digital en los certificados de estudios y en documentos del registro civil, principalmente debido al término de los plazos para la conversión digital. En este sentido, los documentos legales procedentes de algunos estados no se admitían en otros estados al no contener los códigos de verificación correspondientes.

1.5.4. Normas y Leyes internacionales

Existen un conjunto de normas y leyes internacionales aplicables a nuestro país, en particular cuando información de otros países es compartida con entidades nacionales. A continuación describimos algunas de estas normas.

1.5.4.1. GDPR

El Reglamento General de Protección de Datos [38] es una reglamentación europea que habla acerca de los derechos ARCO, originalmente: Acceso, Rectificación, Cancelación y Oposición sobre el tratamiento de los datos personales.

Esta normativa, aunque es europea, es aplicable en México si se desea hacer negocios con algún país de la Unión Europea. Además, podría aplicar en México directamente, al ser la unión Europea una institución que protege los intereses de sus ciudadanos en otros países.

En el caso de México, ya existe una ley similar que protege a los ciudadanos, la Ley Federal de Protección de Datos Personales en Posesión de Particulares y su equivalente para Sujetos Obligados, por lo que su cumplimiento -que es obligatorio- cubre en mayor medida esta normativa internacional.

1.5.4.2. ISO27001:2013

El ISO 27001 [39] es un conjunto de lineamientos que especifica los requisitos para establecer, implementar, mantener y mejorar un Sistema de Gestión de Seguridad de la Información (SGSI) conocido en inglés como *Information Security Management System*).

Estos requisitos describen cuál es el comportamiento esperado del Sistema de Gestión una vez que esté en pleno funcionamiento.

El SGSI es la parte del sistema de gestión de la empresa, basado en un enfoque de riesgos del negocio, para establecer, implementar, operar, monitorear, mantener y mejorar la seguridad de la información. Este incluye la estructura organizacional de la empresa, políticas, actividades, responsabilidades, prácticas, procedimientos, procesos y recursos.

La norma ISO/IEC 27001 se divide en 11 secciones más el anexo A. Las secciones 0 a 3 son introductorias (y no son obligatorias para la implementación), mientras que las secciones 4 a 10 son obligatorias, lo que implica que una organización debe implementar todos sus requerimientos si quiere cumplir con la norma. Los 114 controles (técnicos) del Anexo A deben implementarse sólo si se determina que corresponden en un documento denominado la Declaración de aplicabilidad (SOA).

1.5.5. Normas nacionales

En México, el SAT permite la figura de Proveedor Autorizado de Certificación (PAC), que es la persona moral que cuenta con la autorización para generar y procesar fuera del domicilio fiscal de quien lo contrate, los comprobantes para efectos fiscales por medios electrónicos y de manera 100 % digital. [41, Regla

2.7.2.1 y Anexo 1-A, ficha 112/CFF], [35, Artículo 29, Fracción IV, Segundo Párrafo]

Para poder ser PAC, se tiene que ser persona moral del Régimen General de Ley. Por ejemplo, persona moral con fines no lucrativos, como las cámaras de comercio e industria, agrupaciones agrícolas, ganaderas, pesqueras o silvícolas, así como los organismos que las reúnan, siempre que el servicio se preste únicamente a sus agremiados. Otra opción es persona moral con la actividad de asociaciones, organizaciones y cámaras de productores, comerciantes y prestadores de servicios, con actividades que incluyan la promoción, representación y defensa de los intereses de sus afiliados (productores agrícolas e industriales, comerciantes y prestadores de servicios), siempre que el servicio se preste únicamente a sus asociados o agremiados. También las dependencias y entidades de la federación, entidades federativas y municipios que por ley estén obligadas a entregar al gobierno federal el importe íntegro de su remanente u operación, siempre que el servicio se preste para certificación de las facturas electrónicas de dichas dependencias o entidades.

Esto viene reglamentado en [41] y [35]. En estos reglamentos se indica que requisitos de seguridad se deben de cumplir. Estos están establecidos la denominada Matriz de Controles para la Revisión de Seguridad para Proveedor de Certificación de Comprobantes Fiscales Digitales por Internet (PCCFDI), que es una variante del ISO 27001, pero ajustada a los requisitos particulares del SAT.

1.6. Gestión de riesgos

La gestión de riesgos es el arte de tratar los retos, de lo que puede llegar a pasar, de una manera simple y ordenada. No pretendemos evitar todos los riesgos

o traerlos a colación, pero puede verse como una oportunidad de clarificarlos y predecirlos, y se tratarán tantos como sea posible. [40]

La esencia de la gestión de riesgos es hacerla un poco de clarividente basándonos en lo que sabemos y en cómo se ha manejado anteriormente. Está asociado a encontrar lo que otros saben y han manejado para crear un entendimiento más amplio posible sobre los riesgos y cómo lidiarlos.

1.6.1. ¿Porqué es importante la gestión de riesgos?

Nos ayuda a identificar los errores y aciertos que puedan afectar el éxito de proyectos y de la organización, crea sistemas de alerta temprana, nos ayuda a enfocarnos a lo que hay que atender primero, genera estrategias para alinear los riesgos con la cultura organizacional, resuelve problemas antes de que pasen, y crea las mejores prácticas para el futuro.

La organización establece sus parámetros de lo que es importante preocuparse y de qué no. Los accionistas clave identifican amenazas y oportunidades, clarifican categorías, causas y efectos. Los riesgos son evaluados por su impacto individual y su impacto en la organización como un todo, a través de criterios objetivos y subjetivos, utilizados en una variedad de herramientas.

1.6.2. Plan de Gestión de Riesgos

Riesgo es lo que puede pasarnos, bueno o malo, y que influenciará en cómo nuestros proyectos, trabajo o nuestras vidas puedan acabar. La administración del riesgo es el acercamiento basado en procesos que te lleva a través de la incertidumbre de esos eventos.

Es importante diferencias entre hacer frente a los problemas a nuestro alrededor, y el gestionar estos problemas. La gestión del riesgo trata con aspectos

impredecibles de una manera predecible. Es cuestión de clarificar qué es lo que puede ir mal (o bien) y tener un plan para evitarlos, vivir con ellos o minimizarlos. Cada organización, proyecto o cada individuo tiene su propia cultura de riesgos: es el proceso que la gente sigue. La cultura de riesgo describe que es lo que se va a tolerar y lo que no se va a tolerar.

De un evento que pueda afectar los objetivos del proyecto o de la organización, hay que considerar su probabilidad de ocurrencia y su impacto si llegase a pasar. Para identificar el impacto de un riesgo se pueden utilizar términos medibles como: tiempo, costo, requerimientos o satisfacción del cliente. Además, en el caso de la seguridad de la información, hay que considerar si ésta afecta a los servicios básicos que hayamos definido, como son, confidencialidad, integridad y disponibilidad de la información y en los valores esperados por nuestros clientes, en función de algún contrato legal que tengamos o incluso alguna normativa nacional o internacional, como las ya mencionadas en la sección anterior.

1.6.2.1. Objetivos de la Gestión de Riesgos

Los objetivos de la gestión de riesgos son los objetivos que el proceso idealmente debería de alcanzar, pueden ser oportunidades a tomar, riesgos a gestionar o minimizar o tolerancias a no exceder. Los obejtivos pueden ser tan simples como no ser *hackeado* este año o tan complejos como objetivos detallados de contención de ataques a mediano plazo.

Los objetivos definen lo que a la organización sinceramente le preocupa en cuanto a riesgos. Una manera de definir los objetivos es a través de preguntas sencillas como: ¿Qué condiciones se deben de alcanzar para que esto sea considerado un éxito?, ¿Qué condiciones se consideran totalmente inaceptables en

términos de este esfuerzo?, ¿Hay algún cambio en los parámetros que puedan alterar las respuestas de las preguntas anteriores?.

Cuando se definen objetivos de gestión de riesgos, hay que establecer tolerancias, definir umbrales y puntos de quiebre para tomar acciones.

1.6.2.2. Identificación, Apreciación y Tratamiento de riesgos

Una vez definidos los objetivos de gestión de riesgos, procedemos a identificar los riesgos. Para esto, se deben de identificar los activos de la empresa correspondientes a la información, los activos primarios son la información y los procesos y actividades del negocio, los activos de apoyo son el *hardware*, *software*, la red, el personal, el sitio, la estructura organizacional, etc.

Hay que identificar los flujos de información en los diversos departamentos y procesos, clasificar su importancia para el negocio en los términos de los servicios básicos de seguridad de la información (confidencialidad, integridad y disponibilidad) de acuerdo a los objetivos previamente establecidos. Posteriormente, identificar qué es lo que puede afectar a los mismos, mediante diversas técnicas administrativas, tales como: Llúvia de ideas, grupos de control, reuniones de expertos y el conocimiento que está disponible tanto en la literatura formal como en los foros de discusión de internet.

Una vez identificados los riesgos, mediante la herramienta de Análisis del Modo y Efecto de Fallas (AMEF), podemos dar un apreciación a los mismos. Esto es, definir la consecuencia del riesgo, la probabilidad de que suceda, el valor del activo afectado y su impacto en el negocio de manera objetiva y medible para poder clasificar los riesgos.

Una vez clasificados, la organización decide a cuáles dar prioridad y se establecen planes de acción para su tratamiento. Esto incluye cómo evitarlos, qué

hacer si pasa el riesgo, a quién llamar o cómo ignorarlos. Estos planes de acción se deberán de revisar a intervalos regulares, ya que el contexto es cambiante y la probabilidad o impacto de un riesgo puede variar con el tiempo, por lo que se puede dar una prioridad baja a un riesgo que anteriormente era crítico o al revés.

Capítulo 2

Ciberseguridad en México

Resumen

La educación en ciberseguridad es el punto neurálgico del proceso de inserción en la sociedad de las nuevas generaciones que hacen uso constante de las tecnologías de la información. De ahí que el sistema educativo debiera brindar no solo conocimientos elementales, sino también guiar en el desarrollo de aquellas habilidades que permitían enfrentar la vida tomando en cuenta los retos del ciberespacio de cada época.

En este capítulo se describen algunas líneas de investigacion de la ciberseguridad que son desarrolladas en México. Posteriormente se listan las universidades tanto públicas como privadas en dónde es posible formar recursos humanos especializados en ciberseguridad.

2.1. Líneas de Investigación en México

Hablar de ciberseguridad hace referencia a un conjunto de técnicas que pueden ser agrupadas en diferentes líneas de investigación, entre las que destacan: criptografía, criptoanálisis, internet de las cosas, seguridad en infraestructura, seguridad en sistemas operativos, *software* malicioso y seguridad en infraestructura crítica, por citar algunas. La comunidad científica de México, de la mano de sus respectivos grupos de investigación han encaminado su esfuerzo principalmente en tres líneas de investigación, siendo estas: criptografía, seguridad en infraestrcutura y seguridad en el internet de las cosas. La descripción correspondiente a cada línea se da a continuación.

2.1.1. Criptografía

La criptografía es una ciencia que combina ciertas áreas de las matemáticas con las ciencias de la computación. Esta ha jugado un papel muy importante en la ciberseguridad, enfocándose en la protección de la información a través de cuatro servicios de seguridad: confidencialidad, integridad, autenticación y no repudio [42]. Históricamente hablando, ahora puede clasificarse en: clásica, moderna, cuántica y post-cuántica.

2.1.1.1. Clásica

La criptografía clásica tuvo sus inicios aproximadamente hace más de 4000 años, con el primer registro de la escritura egipcia. En aquel entonces, las sustituciones y las permutaciones eran muy utilizadas para transformar la información [43]. Ejemplo de este tipo de transformaciones es la escítala de los espartanos o la escritura de algunas civilizaciones como los egipcios o los babilonios [44].

La segunda guerra mundial marca un parte aguas para la criptografía ya que aquellos años se dieron avances a pasos agigantados, dando pie al desarrollo de máquinas mecánicas y electromagnéticas para la protección de la información. Un ejemplo de ello se observa con el diseño de la máquina Enigma por parte de los alemanes. A la par, el científico y matemático inglés Alan Turing, fue capaz de desarrollar una máquina para descifrar los mensajes creados por la máquina Enigma, dando inicio al desarrollo de las computadoras, marcando una nueva era criptográfica, la criptografía moderna [43].

2.1.1.2. Moderna

La siguiente gran revolución aparece a finales de los años 60, con el desarrollo de la criptografía de llave pública. Hasta entonces todos los cifradores eran simétricos, en el sentido que las partes debían intercambiar una llave con la cual se cifra y descifra la información. De ahí que la criptografía de llave pública o asimétrica utiliza dos llaves, una de ellas, como su nombre lo dice, es conocida por el público o aquellos receptores con quienes se quiera preservar un servicio de seguridad, y la otra, llamada privada, es solamente conocida por el dueño del par de llaves publico/privado [42].

2.1.1.3. Cuántica

En plena estandarización de los algoritmos pertenecientes a la criptografía moderna, por parte del NIST (*National Institute of Standards and Technology*) [45], es que en los años 70's se tienen las primeras ideas relacionadas con la criptografía cuántica, siendo hasta los 80's cuando se muestran las publicaciones iniciales de nuevos protocolos, los cuales basaban su seguridad en los principios de la mecánica cuántica, como el principio de la incertidumbre o el principio de

la superposición [46]. De ahí, que en los años 90's, los algoritmos criptográficos modernos empezaron a verse afectados por la computación cuántica, lo que con el paso de los años, dio pie al diseño de algoritmos post-cuánticos que a la fecha se encuentran concursando hacia la estandarización por el NIST.

2.1.1.4. Post-cuántica

En los años 90's el profesor de matemáticas aplicadas del MIT Peter Shor, propone un algoritmo cuántico que descompone un número entero N en factores primos en tiempo polinomial. Su implementación podía llevarse a cabo de manera clásica o utilizando circuitos cuánticos [47], el cual basa su potencia en determinar el periodo de una función, para que de esta manera se puedan encontrar factores primos para un entero. De ahí que un ordenador cuántico con un número suficiente de qubits que ejecuten el algoritmo de Shor, podría utilizarse para romper algoritmos modernos de llave asimétrica. De hecho, a la fecha, el algoritmo de Shor y el algoritmo de Grover, creados durante a época de la criptografía cuántica, han podido comprometer algunos de los algoritmos utilizados en la criptografía moderna hasta estos días.

La existencia de este tipo de algoritmos ha provocado dudas relacionadas con el uso de los algoritmos criptográficos modernos, ya que su seguridad quedaría expuesta y como consecuencia, también la de aquellas soluciones que los utilicen [**?**]. De ahí, que la comunidad científica, ha invertido mucho tiempo en el diseño de nuevos algoritmos criptográficos resistentes a ataques efectuados con computadoras cuánticas, mejor conocidos como algoritmos post-cuánticos. Lo anterior, tomando en cuenta algunas consideraciones, entre las que destacan: (1) que el cómputo cuántico usa reglas que son poco intuitivas, y (2) que

algunos cálculos, como la transformada cuántica de Fourier, se pueden realizar exponencialmente más rápido que con el cómputo clásico.

Cuando los algoritmos de la criptografía moderna empiezan a ser comprometidos por el cómputo cuántico, surge la criptografía post-cuántica, que hace referencia a los algoritmos diseñados para resistir ataques de computadoras cuánticas. La búsqueda de problemas matemáticos difíciles incluso para computadores cuánticos, ha llevado al surgimiento de varias ramas dentro de la criptografía pos-cuántica como por ejemplo, (a) algoritmos basados en lattices, (b) algoritmos basados en ecuaciones multivariables, (c) algoritmos basados en curvas elípticas isogeneas supersingulares, (d) algoritmos basados en hashes, y (e) algoritmos basados en códigos.

2.1.2. Seguridad en infraestructura

El intercambio de información entre personas, organizaciones, dispositivos físicos y digitales sobre las redes de datos se ha vuelto un punto crítico dado que día con día se transmite más información sensible y privada en medios como el internet.

Para mantener un nivel adecuado de seguridad en nuestra infraestructura, es necesario seguir las mejores prácticas de seguridad, tanto perimetral como dentro de una red local, e incluso de prácticas comunes relacionadas con la protección interna de un servidor.

Un experto en seguridad en infraestructura, deberá de poder configurar canales de comunicación seguros en una red local, una red abierta, implementar protección por medio de *Firewall* y sistemas de detección y protección de redes, asimismo, realizar pruebas de penetración a aplicaciones web es deseable.

Respecto a la manera de proteger la infraestructura, existen diversas metodologías para hacer el "*hardening*" [1] de un sistema. En particular, la recomendación es seguir la normativa de certificación ISO 27001, ya que esta permite mantener un control documentado de todas las operaciones dentro de la institución, así como el proceso de apreciación y el de tratamiento de riesgos. La denominada "parte técnica" de la norma se puede cubrir con otras certificaciones de seguridad en infraestructura que los principales proveedores de soluciones en servidores de red no permiten. Adicionalmente, una rama de la seguridad en infraestructura tiene que ver con el *hackeo* ético de los equipos. Éste busca atacar a los sistemas informáticos en la búsqueda de vulnerabilidades con la finalidad de subsanar las debilidades encontradas.

2.1.3. Internet de las cosas

Existen varias estrategias verticales de la Internet de las Cosas (IoT): consumidor, comercial y las formas industriales del internet para el Internet de las Cosas, dependiendo de su audiencia objetivo, requerimientos técnicos y estrategias de solución.

El consumidor tiene el mercado más alto de visibilidad con las casas inteligentes, los dispositivos personales de monitoreo de salud y rendimiento atlético, los dispositivos de entretenimiento integrado, así como los monitores personales en los coches. De manera similar, el mercado comercial tiene una alta visibilidad, pero en servicios que abarcan la parte financiera y de productos de inversión, tales como banca, seguros y servicios financieros y el comercio electrónico, los cuales se enfocan a la historia del consumo, desempeño y valor. En el caso de

[1] Proceso de asegurar un sistema reduciendo sus vulnerabilidades y garantizar su correcto funcionamiento

la IoT industrial, se incluye dentro de su mercado los negocios pequeños, medianos y grandes e incluyen aplicaciones de producción energética, manufactura, agricultura, cuidado de la salud, transporte, logística, aviación, viajes y más. Previo a la pandemia que comenzó en el 2019, se tenía una proyección financiera de crecimiento muy importante para este rubro.

Del lado de la seguridad en el IoT, existe una -arraigada- creencia de que estos dispositivos son inseguros. Gracias a su popularidad y a que están diseñados para su uso masivo, se distribuyen con opciones de configuración que faciliten su uso, al grado de que solamente se necesite presionar un botón y que haya un celular cerca para que comiencen a trabajar. Esta facilidad también ha hecho que sean una plataforma deseable para los ciberataques.

En agosto de 2016, los ataques iniciales del malware Mirai produjeron alrededor de 1 TB por segundo de tráfico de internet a los servidores de DNS Dyn, que produjeron grandes transtornos en diversos sitios de internet, incluyendo a algunos de los más visitados a nivel mundial [**?**], y de hecho, México fue el país más afectado en el 2017 [**?**]. Este malware Mirai ha expuesto los riesgos asociados a la infraestructura de internet, además de que permite la generación de variantes que se pueden ejecutar y que hacen un poco más difícil su detección, mitigación o eliminación [**?**]. Incluso, se ha creado un modelo de negocios denominado *malware-as-a-service*, en donde los usuarios se suscriben a una red de ataque y los autores reciben una comisión por éstos, así como del pago correspondiente a la suscripción. [**?**]

Más allá de los problemas de ingeniería de software de los dispositivos IoT, tenemos el caso de que su poder de cómputo generalmente está asociado a una falta de interés por implementarles capas criptográficas en sus protocolos de comunicación. De acuerdo a Scott [**?**], dado el incremento en poder de cómputo

de los dispositivos embebidos y de IoT, no hay excusa para no aplicar los esquemas criptográficos contemporáneos a los protocolos de seguridad y comunicación en dichos dispositivos. Incluso, existe investigación acerca de cómo implementar la criptografía poscuántica en algunos dispositivos embebidos de gama alta. [**?**]

2.2. Educación y ciberseguridad en México

A continuación se listan las universidades tanto públicas como privadas en donde es posible formar recursos humanos especializados en ciberseguridad en México.

2.2.1. Centro de Investigación y de Estudios Avanzados

El Centro de Investigación y de Estudios Avanzados (CINVESTAV) en su Unidad Zacatenco [49] tiene un grupo de criptografía y seguridad informática. Este grupo puede verse como el más antiguo de su tipo en el país. Sus primeros antecedentes se remontan a finales de la década de los ochentas del siglo pasado, pero fue hasta después del año 2000, cuando el grupo adquirió una mayor presencia, mediante la contratación de varios especialistas que permitieron un crecimiento importante en su quehacer de investigación, docencia y formación de recursos humanos.

Actualmente, el grupo de investigación del CINVESTAV cuenta con la participación de cinco profesores investigadores que trabajan de tiempo completo en las disciplinas de criptografía y seguridad informática, junto con un número considerable de investigadores asociados, que colaboran e interactúan activamente con los miembros del grupo.

El grupo tuvo un papel muy activo en la fundación de la conferencia "Latincrypt" y de la Escuela de seguridad "ASCrypto" en 2010. A partir de entonces, *The International Conference on Cryptology and Information Security in Latin America* (Latincrypt) y la *Advanced School on Cryptology and Information Security in Latin America* (ASCrypto), se han convertido en los dos eventos académicos más relevantes en Latinoamérica en las áreas de criptografía y seguridad informática.

2.2.2. Instituto Politécnico Nacional

El grupo de Seguridad del Instituto Politecnico Nacional (IPN) [50] tuvo sus inicios en 1998, casi de la mano de la fundación de la Sección de Posgrado de la Escuela Superior de Ingeniería Mecánica y Eléctrica, Unidad Culhuacan (ESIME Culhuacan).

Los primeros esfuerzos asociados a este grupo de seguridad, empezaron a dar sus pequeños frutos en 1999, con la puesta en marcha de un seminario de actualización con opción a titulación de nivel licenciatura. Posteriormente, en 2004, este grupo se expandió e inició tareas dirigidas hacia la creación de un programa de posgrado de nivel especialidad, el cual en el 2006 quedó registrado bajo el nombre de Especialidad en Seguridad Informática y Tecnologías de la Información, convirtiéndose en el primer programa de posgrado de seguridad en México, dirigido hacia el público civil, quién podía armar su mapa curricular enfocandose en tres áreas principales: Protección de la información, sistemas criptográficos y protección de sistemas informáticos.

Dada la cantidad de aspirantes interesados en dicho programa, en ese mismo año, se dá inicio a las tareas encaminadas a la creación de un nuevo segundo programa de seguridad, pero ahora de nivel Maestría, quedando registrado bajo

el nombre de Maestría en Seguridad y Tecnologías de la Información en 2008, ofertando tres diferentes líneas de desarrollo, siendo estas: identificación y derechos de autor, protección de sistemas de información y canales de comunicación seguros.

A partir de 2011 y con la finalidad de seguir creciendo como investigadores en el área de la Seguridad, algunos de los miembros fundadores de citados programas de la ESIME, migran hacia el Centro de Investigación en Computación (CIC), y en 2013 fundan el Laboratorio de Ciberseguridad, enfocando sus esfuerzos en diferentes líneas de investigación, entre las que destacan: seguridad en el ciberespacio, seguridad en el Internet de las cosas, criptografía, algoritmos evolutivos para ciberseguridad y biometría, dedicandose a realizar investigación básica, aplicada y desarrollo tecnológico, formación de recursos humanos altamente especializados y transmisión del conocimiento, en materia de seguridad del ciberespacio, la información electrónica y las tecnologías de información y comunicaciones que lo soportan, para contribuir al desarrollo y bienestar de la sociedad.

2.2.3. Universidad Iberoamericana

En 2017 la Universidad Iberoamericana (IBERO) [51] comenzó a ofertar el programa de Especialidad y Maestría en seguridad de la información. Su objetivo principal es aplicar metodologías y técnicas de protección de la información, con sustento ético, basadas en el uso de diversas herramientas de seguridad informática, dando un enfoque encaminado hacia la normatividad sobre privacidad de datos y medidas de ciberseguridad.

El programa de especialidad en seguridad de la información prepara profesionales con habilidades para integrar los temas de auditoría, riesgos, criptogra-

fía y aspectos forenses en tecnologías de la información. Cabe destacar que para ingresar al programa de maestría, primero se debe aprobar la especialidad con duración de un año para posteriormente estudiar dos semestres de la maestría.

2.2.4. Universidad La Salle

La finalidad del programa de Maestría en Ciberseguridad de la Universidad La Salle [52] es formar maestros con los conocimientos, las habilidades y las actitudes que les permitan diseñar, dirigir, ejecutar y evaluar iniciativas, proyectos y planes estratégicos de ciberseguridad, acordes con las arquitecturas empresariales y alineados con los objetivos de las organizaciones. Todo esto, atendiendo tanto principios de actuación ética como estándares y regulaciones nacionales e internacionales. Su creación fue en 2017.

Al igual que la IBERO, se tiene un programa en el que es posible inciar una especialidad y después realizar la maestría. La duracion del plan de estudios es de 2 años, dividido en 6 cuatrimestres, en los cuales se tocan temas como: seguridad en redes, criptografía, *hackeo* ético, gestión de incidentes y análisis forense y auditoría de la seguridad, todo esto sin descuidar temas de factores Humanos en la seguridad de la información y algunos temas de normatividad.

2.2.5. Universidad Tecnológica de México

La Maestría en Seguridad de Tecnología de Información de la Universidad Tecnológica de México (UNITEC) [53] se diseño en el 2017 para cursarse en cuatrimestres. Tiene un enfoque práctico que permite combinar trabajo y estudio. Su modalidad es presencial y es impartida en el Campus Marina y Sur. Este programa esta dirigido a pasantes o titulados en licenciaturas o ingenierías de áreas afines a TI's, comunicaciones o electrónica. Dentro del plan de estudios se

puede encontrar materias como: arquitectura de la seguridad, controles criptográficos de seguridad, *hacking* ético y analisis forense, seguridad en dispositivos móviles y telefonia, normatividad y legislación de IT, entre otras. La duración del programa es de 1 año 8 meses.

2.2.6. Instituto Nacional de Astrofísica Optica y Electrónica

El Instituto Nacional de Astrofísica Óptica y Electrónica (INAOE) [54] ofrece dos maestrías en las que es posible estudiar temas relacionados con Ciberseguridad: la Maestría en Ciencias Computacionales y la Maestría en Ciencias en Tecnologías de Seguridad. El primer programa es de dedicación exclusiva y pertenece al padrón de programa nacional de posgrados de calidad (PNPC) en la categoría de competencia internacional, por lo que puede ofrecer a sus estudiantes becas otorgadas por el CONACyT. Los estudiantes cubren materias que les ayudan a consolidar sus conocimientos en Ciencias Computacionales y pueden cursar materias para especializarse en Criptografía y Seguridad, línea de investigación que se trabaja desde hace más 15 años en este programa.

El segundo programa comenzó en el 2018. Es un programa con la Industria por lo que es de modalidad semipresencial. Las clases se imparten los viernes de manera presencial y el resto de la semana se trabaja por medio de una plataforma digital. Este posgrado también pertenece al PNPC en la categoría de posgrados de reciente creación, de igual forma cuenta con becas CONACyT que permiten que el estudiante unicamente cubra la mitad de su costo. A diferencia de la maestría en Ciencias Computacionales, en este programa, todas las materias ofertadas son del área de Seguridad. Su objetivo es formar maestros en ciencias con los conocimientos, habilidades y aptitudes que les permitan liderar, anali-

zar, diseñar, aplicar y evaluar ideas, proyectos y planes estratégicos de seguridad cibernética y diseño de sistemas optrónicos. Esto, conforme a las arquitecturas empresariales y alineadas con los objetivos de las organizaciones. Todo ello ligado a los principios de actuación ética como estándares y regulaciones nacionales e internacionales en un marco legal de actuación.

2.2.7. Tecnológico de Monterrey

En el 2018 de la mano de empresas como: Cisco, Citibanamex, Deloitte, IBM, Thales y la Universidad de Texas, San Antonio (UTSA), el Tecnológico de Monterrey (Tec) [55] creó el Tec Cybersecurity Hub, un centro de operaciones de ciberseguridad que genera espacios para que investigadores, profesores, alumnos e inversionistas se sumen a la misión de investigar, capacitar y concientizar sobre las amenazas cibernéticas, teniendo como resultado, la formación de recursos humanos de calidad.

Un año más tarde, se crea el programa de Maestría en Ciberseguridad del Tec [19]. Su objetivo es formar profesionistas y agentes de cambio en las organizaciones, de tal forma que sean capaces de hacer innovación, desarrollo tecnológico y transferencia de tecnología en las áreas de la ciberseguridad. Así como liderar y gestionar una oficina de ciberseguridad. Se imparte en el Campus Monterrey, Santa Fe y Guadalajara. Su periodo es trimestral y tiene duración de 1 a 3 años.

Finalmente en el 2020, el Tecnológico de Monterrey inició el Diplomado en Tecnologías de Ciberseguridad – Live, el cual integra servicios, mecanismos y controles para contribuir a las mejores prácticas de la arquitectura, operación y gobierno de la ciberseguridad. Todos ellos alineados con los objetivos de la organización. Esta modalidad de educación continua y de impartición sincrónica se lleva a cabo mediante un aprendizaje interactivo, virtual y experiencial.

2.2.8. Universidad Autónoma de Nuevo León

La Universidad Autónoma de Nuevo León (UANL) [56] dentro de su facultad de Ciencias Físico-Matemáticas ofrece la Maestría en Ingeniería en Seguridad de la Información. Este programa se divide en 6 semestres con materias que van desde criptografía, pasando por seguridad en base de datos y sistemas de control de accesos hasta *hackeo* ético, sin descuidar la parte norvativa que conlleva la Ciberseguridad.

2.2.9. Universidad en Internet

La llamada Maestría en Seguridad Informática de la Universidad en Intertet (UNIR México) [57] es un programa con validéz oficial y 100 % en línea. Permite obtener el título en 18 meses. Su finalidad es convertir al estudiante en un profesional preparado para cubrir las necesidades actuales de las empresas, tanto del sector público como privado. Esto, a través del estudio de técnicas de protección ante vulnerabilidades de sistemas operativos, *software*, bases de datos, sistemas *web*, así como su posible repercusión dentro de la organización. Su plan de estudios abarca la seguridad de la información desde el punto de vista legal, técnico y de gestión.

Al concluir esta maestría, el estudiante es acreedor al doble titulo, uno mexicano y uno europeo, este último es otorgado por la Universidad Internaciónal de la Rioja, España.

2.2.10. Centro de Estudios Superiores Navales

Desde el 2005 el Centro de Estudios Superiores Navales (CESNAV) [58] ha ofertado un programa de posgrado de nivel maestría cerrado exclusivamente pa-

ra personal de las Fuerzas Armadas de México y otros paises amigos en convenio internacional, así como para elementos del Consejo Nacional de Seguridad de México.

Su programa de Maestria en Seguridad de la Información está constituida por cuatro módulos con duración total de un año escolarizado. El objetivo de su plan de estudios es que los alumnos conozcan los fundamentos teórico-prácticos que implican la seguridad de la información, de tal manera que puedan resolver problemas actuales y reales en materia de ciberseguridad. Este programa ofrece dos áreas de interes: Administración de la seguridad de la información y el área operativa. La primera tiene como principales funciones administrar el desarrollo y aplicación de los sistemas de seguridad de la información, así como aplicar estas herramientas para apoyo en la seguridad, además de analizar y evaluar distintas propuestas de seguridad en la información de acuerdo a las políticas institucionales, entre otras. Desde el punto de vista del lado operativo, entre sus funciones tiene el proporcionar soporte técnico y mantener la seguridad informática, así como aplicar técnicas de cómputo forense, criptografía y tecnicas de inteligencia de señales, por mencionar algunas.

2.3. Diplomados de Ciberseguridad en México

2.3.1. Universidad Nacional Autonoma de México

La carrera de Ingeniería en Computación de la Universidad Nacional Autonoma de México (UNAM) [59] preocupada por atender necesidades presentes y futuras en el campo de la computación y la información, generó un Diplomado en redes y seguridad, el cual tiene como objetivo dar a los asistentes, los conocimientos y capacidades teórico-prácticas que se requieren para la ciberseguridad,

así como impulsar y promover la cultura de la ciberseguridad en un marco ético y profesional.

Este diplomado ofrece 6 campos de conocimientos que se alcanzan a lo largo de 240 horas. Los campos son: Área humana, área estratégica, acción y reacción, ciberseguridad y el mundo industrial, ámbito legal y financiero y el campo de liderazgo en la ciberseguridad. Cada uno de los 6 campos tiene duración y objetivos diferentes, que van desde la identificación de las caracteristicas del ser humano en el ciberespacio y su relación con la ciberseguridad, pasando por las técnicas y métodos para la detección y prevencion de amenazas a la ciberseguridad como: la investigacion de *malware*, ciberseguridad en la nube e IoT y analisis forense, hasta llegar a la construccion y vigilancia de la ciberseguridad y obtener Funciones y habilidades del CISCO.

El diplomado se toma los dias viernes y sabados en el Laboratorio de Redes y seguridad de la Facultad de Ingeniería en la UNAM.

2.3.2. Universidad del Valle de México

La Universidad del Valle de México (UVM) [60] oferta el diplomado de Ciberseguridad y ciberdefensa con el objetivo de profundizar sobre los principales elementos de identificación, protección, detección, respuesta y recuperación ante una amenaza en ciberseguridad y alinear los recursos que ofrecen las tecnologías de la información con los objetivos de negocio o institucionales.

El programa tiene una duración de 128 hrs, que se dividen en 9 módulos que abarcan distintos temas relacionados a la ciberseguridad. Su modalidad es presencial y se imparte en el Campus Coyoacán.

2.3.3. Universidad Anahuac

El diplomado en línea en ciberseguridad de la Universidad Anahuac [61] tiene un enfoque holístico y busca dotar a los participantes de las competencias digitales necesarias para combatir el creciente problema en materia de seguridad informática.

Este diplomado está dirigido principalmente a profesionales encargados de la operación y gestión de sistemas digitales, consultores de seguridad informática, técnicos vinculados a las TIC y personas que hagan uso de las TIC.

El plan de estudios prepara profesionales capaces de desempeñarse en empresas dedicadas a la operación de sistemas informáticos y de seguridad informática asi como áreas dedicadas al manejo de los sistemas informáticos en el gobierno. Tiene duración de siete meses y un total de 125 hrs de instrucción estimada en línea, dividida en 5 módulos de estudio con temas enfocados en: Introducción a la ciberseguridad, riesgos actuales en ciberseguridad, informática forense, seguridad de la información y legislación nacional e internacional en ciberseguridad.

Capítulo 3

Criptografía moderna: cifrado y hash

Resumen

La criptografía es una ciencia que a lo largo de los años ha creado su propia historia. Tuvo sus inicios en la cultura egipcia hace unos 4000 años. Durante las guerras mundiales jugó un papel sumamente importante. En aquel entonces solo la utilizaban los militares, el servicio diplomático y el gobierno, ya que su finalidad era proteger estrategias y secretos nacionales.

Tiempo después, el surgimiento de las computadoras y los sistemas de comunicación llegó de la mano de la necesidad del sector privado de proteger la información digital.

Históricamente la criptografía puede clasificarse en clásica, moderna, cuántica y postcuántica. En este capítulo se presenta una vista hacia la criptografía moderna, enfocándose en diferenciar la criptografía simétrica de la asimétrica y se termina el capítulo describiendo las funciones hash.

3.1. Criptografía de llave simétrica

Imaginemos que Alicia desea comunicarse con Beto utilizando la internet, pero saben que es un canal de comunicaciones inseguro ya que hay entidades que tienen acceso a toda la información que viaja a través de ella. Desde luego Alicia y Beto no quieren que nadie se entere del contenido de su comunicación. Para ello pueden utilizar *criptografia de llave simétrica*, que les permitirá convertir los mensajes a un formato que resulte ininteligible para cualquier persona que los vea, a excepción de aquellos que conozcan la llave (o clave).

El cifrador lo podemos ver como una caja (función) que toma como entrada un mensaje en claro y mediante un procedimiento que incluye el uso de la llave, lo convierte a un texto cifrado. Luego ese texto cifrado – que es inteligible – viaja a través del canal inseguro, y aunque cualquiera puede verlo durante su trayecto, nadie puede obtener el mensaje en claro original.

Cuando el mensaje cifrado llega hasta las manos de Beto, él utilizando la misma llave es capaz de descifrar el mensaje. Es decir, Beto utiliza la función inversa y con la llave es capaz de recuperar la información original que Alicia le mandó. La idea se ilustra con la Figura 3.1.

Además de utilizarse para cifrar información que viajará de un usuario a otro, también es posible mantener información de manera confidencial, así por ejemplo un arhivo puede cifrarse con una llave y guardarse en un dispositivo de almacenamiento, y únicamente quién posea la llave será capaz de descifrar el archivo.

Se le llama *llave simétrica* porque existe una simetría entre la llave que se usa para cifrar y la que se usa para descifrar (de hecho es la misma llave).

Figura 3.1: Esquema de comunicación con criptografía de llave simétrica entre Alicia y Beto.

Desde luego debe notarse que antes de poder utilizar este esquema, es necesario que Alicia y Beto acuerden la llave que aplicarán, para ello deben utilizar un canal de comunicaciones que sea seguro para mantener la secrecía de la llave.

Por ejemplo, podrían reunirse personalmente para hacer el acuerdo de las llaves, o pueden valerse de otros esquemas criptográficos que permiten el intercambio de llaves, como el llamado protocolo de Diffie-Hellman, u otros basados en Curvas Elípticas (ver en las siguientes secciones).

Los primeros esquemas de llave simétrica surgieron en la década de los años 70 del siglo XX, y hasta nuestros días se usan en las comunicaciones modernas día con día. De hecho, existe toda un área de investigación en criptografía que analiza los esquemas existentes, propone mejoras y nuevos algoritmos.

3.1.1. Cifradores de flujo y de bloques

Existen principalmente dos tipos de cifradores de llave simétrica: los **cifradores de flujo** y los **cifradores por bloques**.

- Los **cifradores de flujo** reciben los datos de entrada, un byte cada vez, y mientras los van recibiendo los cifran y arrojan la salida que les corresponde. Una función muy importante en estos cifradores es el Generador

de Números Pseudo Aleatorios, cuya salida es un flujo de números que dan la apariencia de ser aleatorios y que se usan para generar la llave del cifrador-descifrador. Algunos ejemplos de cifradores de flujo son RC4 [62] y ChaCha20 [63].

- Por otro lado, en los **cifradores por bloques** se particiona el mensaje que Alicia desea enviar dividiéndolo en bloques de un tamaño fijo, luego cada bloque entra al cifrador y es procesado para obtener su salida. A continuación los bloques cifrados viajan por el canal inseguro y al llegar a su destino, el receptor Beto descifra cada uno de ellos utilizando la misma llave que Alicia aplicó.

  Existen diferentes maneras de configurar los bloques que corresponden a un mismo mensaje, de tal forma que Beto sea capaz de obtener el mensaje original sin comprometer ninguna información ni caer en errores; a esas configuraciones se les conoce con el nombre de *modos de operación*.

  Algunos ejemplos de cifradores por bloques famosos son DES (*Data Encryption Standard* [64], IDEA (International Data Encryption Algorithm) [65], AES (*Advanced Encryption Standard*) [66] y Blowfish [67].

  Se considera que un cifrador de llave simétrica por bloques es competitivo, si en primer lugar ofrece la confidencialidad de la información y si además es eficiente. La eficiencia suele medirse en términos del tamaño de llave, simplicidad de los algoritmos y velocidad de cifrado y descifrado.

3.1.1.1. Ejemplo: AES

El cifrador por bloques más ampliamente conocido y utilizado es el llamado AES (aunque su nombre original es Rijndael), que es el estándar aceptado por

la agencia de estándares de Estados Unidos (NSA, *National Security Agency*) en el año 2001. Tiene tamaño de bloques de 128 bits y tamaños de llave posibles de 128, 192 ó 256 bits. El algoritmo trabaja realizando corrimientos y sustituciones en una matriz de 4x4 y tiene cuatro funciones principales que se aplican en varias rondas, éstas son: Sumar bytes, recorrer renglones, mezclar columnas, sumar la llave.

El algoritmo de AES ha sido implementado de manera muy eficiente tanto en plataformas de *hardware* como de *software*. Hasta el momento no se conocen ataques que resulten devastadores o que compromentan de manera importante la información que se cifra con este algoritmo.

3.2. Criptografía de llave asimétrica

La criptografía asimétrica marcó la siguiente revolución en la seguridad de la información, al permitir que cada participante use un juego de dos llaves diferentes pero que corresponden de forma única entre sí para el proceso de cifrado y descifrado. Estas llaves son conocidas como *llave pública* y *llave privada*, las cuales se denotan matemáticamente como (p_k, s_k), respectivamente. Es por ello, que la criptografía asimétrica también es conocida como *criptografía de llave pública*. Esta idea se ilustra con la Figura 3.2.

De esta forma, los participantes únicamente dan a conocer su llave pública, y mantienen en secreto su llave privada, facilitando así el intercambio de información que en el caso de la criptografía simétrica, implica un proceso de intercambio de llaves adicional y que puede representar un problema de difícil manejo [68].

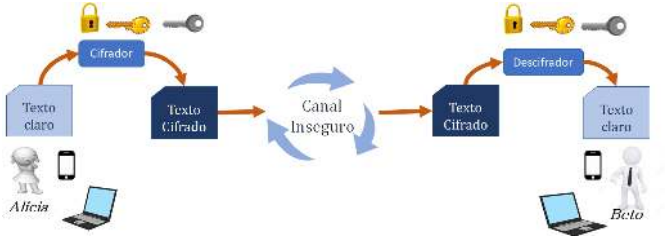


Figura 3.2: Esquema de comunicación con criptografía de llave asimétrica entre Alicia y Beto.

Aunque normalmente se le atribuye el desarrollo de este tipo de criptografía a Whitfield Diffie, Martin Hellman y Ralph Merkle alrededor del año 1976, hay indicios de que pudo haber sido desarrollada en 1972 por James Ellis, Clifford Cocks y Graham Williamson de la oficina de comunicaciones del gobierno británico [69].

La idea de la criptografía asimétrica es que sólo el propietario de la llave privada puede descifrar el texto cifrado que haya recibido. Así, por ejemplo, para cifrar un texto para *Alicia*, el procedimiento implicaría hacer una operación, denotada por $\oplus$, con el mensaje y con la llave pública de Alicia. Es decir:

$$Mensaje \oplus Alicia_{p_k} = Texto\ Cifrado$$

Y para que Alicia recupere el mensaje, necesitaría hacer una operación inversa, denotada por $\otimes$, entre el texto cifrado y su llave privada. Es decir:

$$Texto\ Cifrado \otimes Alicia_{s_k} = Mensaje$$

La seguridad que ofrece la criptografía asimétrica radica en las operaciones $\oplus$ y $\otimes$, ya que su base se encuentra en problemas matemáticos que son considerados difíciles de resolver. Esto es, que el poder de cómputo disponible en la

actualidad no es suficiente para resolver un problema matemático en un tiempo factible. De acuerdo a su relevancia práctica, existen tres problemas matemáticos que dan pie al desarrollo de esquemas criptográficos seguros. De acuerdo con [69], estos son: factorización de números enteros, problema del logaritmo discreto y problema de logaritmo discreto en curvas elípticas.

- **Factorización de números enteros:** Su base está en la dificultad de factorizar números enteros grandes. El algoritmo criptográfico de *RSA*, uno de los más populares de su clase, usa este problema subyacente.
- **Problema del logaritmo discreto:** Existen varios esquemas criptográficos con base en el problema del logaritmo discreto. Entre ellos destaca: El intercambio de llaves de Diffie-Hellman y el cifrado conocido como Elgamal. El problema consiste en calcular un logaritmo de y en base g, donde y y g son elementos de un grupo cíclico finito multiplicativo G^*.
- **Problema del logaritmo discreto en curva elíptica:** Es una generalización del problema del logaritmo discreto, cuyo algoritmo más popular es la firma digital mediante curva elíptica (ECDSA).

3.2.1. El problema de la factorización

El problema de la factorización de define de la siguiente forma: Dado un entero positivo n, encontrar su factorización en números primos. Es decir, hallar una expresión tal que $n = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$ donde p_i son parejas de primos distintos y cada $e_i > 0$ [42].

Este problema es la base del algoritmo *RSA*, que permite cifrar/descifrar un mensaje. Su proceso, en términos generales, se describe a continuación. Se toman dos números primos p y q tan grandes como sea posible, y se calcula su

producto, $n = pq$. Sea e un número entero impar, y que no sea un factor del resultado del producto de $m = (p-1)(q-1)$. Finalmente, sea d un número tal que $e \cdot d \equiv 1$ mód m. La llave pública se compone del par de números (e, n) y la llave privada por el par de números (d, n). Así, el cifrado de un mensaje se obtiene mediante la expresión 3.1.

$$E(M) = M^e (\text{mód} n) \tag{3.1}$$

Por otra parte, el descifrado obedece a la expresión 3.2.

$$D(E(M)) = M^d (\text{mód} n) \tag{3.2}$$

Resolver este problema mediante una computadora clásica resulta una tarea sumamente compleja, que podría tomar incluso años antes de ser completada. Sin embargo, una computadora cuántica tiene la capacidad de hacerlo en un tiempo aceptable mediante el algoritmo de Shor, el cual se centra en las potencias de un número al aplicarles la operación módulo y buscar predecir su *período*, que es el número de veces que se repite la operación antes de que aparezcan los mismos resultados [70]. Por ejemplo, sea x un número al cual se le calculan sus potencias módulo n, donde $n = pq$. Si x no es divisible entre p ni q, entonces la secuencia x mód n, x^2 mód $n, \ldots$ se repetirá con un período θ que será un divisor de $(p-1)(q-1)$. Encontrar el período de esta serie sería entonces el objetivo de una computadora cuántica, que mediante la superposición de estados puede ejecutar múltiples operaciones de forma paralela.

3.2.2. El problema del logaritmo discreto

Entre las técnicas criptográficas que tienen base en el problema del logaritmo discreto se encuentran el *acuerdo de llaves de Diffie-Hellman*, y los esquemas de

cifrado y de firma de ElGamal. Estos pueden definirse de la siguiente forma: Sea G un grupo cíclico de orden n. Sea α un generador de G y sea $\beta \in G$. El logaritmo discreto de *beta* con base en α, denotado por $log_{\alpha}\beta$, es un entero único x, $0 \leq x \leq n-1$, tal que $\beta = \alpha^x$. Luego, el problema del logaritmo discreto consiste en: Dado un número primo p, un generador α de $\mathbb{Z}_p^*$, y un elemento $\beta \in \mathbb{Z}_n^*$, encontrar un entero x, $0 \leq x \leq p-2$ tal que $a^x \equiv \beta(\text{mód}p)$.

La forma inmediata de resolver este problema consiste en una búsqueda exhaustiva, al calcular sucesivamente $\alpha^0, \alpha^1, \alpha^2, \ldots,$ hasta obtener el valor β, lo que implicaría $O(n)$ multiplicaciones, donde n es del orden de α, por lo que resulta costoso e ineficiente para valores grandes de n.

3.2.3. Criptografía de curvas elípticas

La seguridad de la criptografía de curvas elípticas se basa en el problema del logaritmo discreto generalizado, que se enuncia de la siguiente forma: Dado un grupo cíclico finito G de orden n, un generador α de G, y un elemento $\beta \in G$, encontrar el entero x, $0 \leq x \leq n-1$ tal que $\alpha^x = \beta$. Menezes, Okamoto y Vanstone demostraron cómo el problema del logaritmo discreto para una curva elíptica sobre un campo finito $\mathbb{F}_q$ puede ser reducido al problema del logaritmo discreto en una extensión del campo $\mathbb{F}_{q^k}$[42].

Una de las aplicaciones más populares de este problema es, en el algoritmo de firma digital mediante curvas elípticas (ECDSA), que, entre otras cosas, sostiene sistemas como Bitcoin. Geométricamente, una curva elíptica es un conjunto de puntos descritos por la ecuación 3.3.

$$\{(x, y) \in \mathbb{R}^2 \mid y^2 = x^3 + ax + b,\ 4a^3 + 27b^2 \neq 0\} \cup \{0\} \tag{3.3}$$

Las curvas elípticas son grupos, por lo que cumplen la propiedad de cerradura, asociatividad, tienen elemento de identidad y elemento inverso para cada elemento. En particular, el elemento de identidad es el punto al infinito, 0. El inverso del punto P es el punto simétrico respecto al eje x. La adición se define dados tres puntos colineales diferentes de cero que intersecten la curva, P, Q, R, entonces $P + Q + R = 0$. Si sólo hay dos puntos de intersección, es decir, una recta tangente es la que intersecta a la curva, entonces se dice que $P = Q$ y por tanto, $P + P = R$ o bien, $2P = R$, lo que se denomina *doblado de un punto*. La adición y el doblado de puntos permiten definir multiplicaciones escalares para curvas elípticas tales que $xP = R$, donde x es un escalar, P es un punto tangente a la curva y R es el resultado de sumar P a sí mismo x veces.

Para ejecutar una firma con curva elíptica, se eligen los parámetros del módulo, los valores a y b de la ecuación 3.3, el punto generador P y su orden de acuerdo al estándar para la criptografía de grupos eficiente. Luego, la llave privada es un entero que se elige al azar entre los valores 1 y el orden seleccionado. La llave pública se calculará por medio del producto escalar entre el entero que representa la llave privada y el punto generador P, lo que da como resultado un punto (x, y). Aunque resulta fácil calcular la llave pública a partir de la privada, hacerlo en sentido inverso es computacionalmente difícil, por lo que se considera una función de una sola vía [71].

3.3. Funciones hash

Una *función hash* es una función que se utiliza para mapear información digital de cualquier longitud a datos digitales de tamaño fijo. Los valores devueltos por este tipo de función se denominan digesto, valores resumen, o simplemente

Figura 3.3: Esquema general del funcionamiento de una función hash

valores *hash*. Este digesto es único para un valor de entrada dado, así cualquier cambio en los datos de entrada cambia de manera significantiva el dato de salida [42]. De ahí que el digesto de un dato también se le denomine *huella digital*, ya que estos pueden servir como una cadena que identifica de manera única al dato original. Este proceso se puede observar en la Figura 3.3.

Una *función hash criptográfica* [42] es un tipo de función que poseé ciertas características matemáticas. Esta función es unidireccional ya que dados ciertos datos de entrada estos se relacionan a un digesto determinado, pero dado un digesto es sumamente difícil obtener el mensaje original de donde se obtuvo éste. Los datos de entrada a menudo se denominan mensaje, y el resultado se denomina resumen o digesto del mensaje. Si se desconocen los datos de entrada, es extremadamente difícil reconstruir esos datos conociendo sólo el digesto, haciendo de esta función una *función de sólo ida*. Las funciones hash son parte fundamental de la criptografía moderna.

3.3.1. Estructura básica de una función hash

Dado lo mencionado anteriormente podemos construir la siguiente definición.

Una función hash $h(m)$ es una función computacionalmente eficiente que relaciona cadenas de datos m de cualquier tamaño a cadenas de datos de tamaño fijo x. Para que sea segura, la función hash $h(x)$ debe cumplir además ciertas características:

- $h(m)$ es de longitud fija, independientemente de la longitud de m.
- Dado m, es fácil calcular $h(m)$.
- Dado $h(m)$, es computacionalmente difícil obtener m.
- Dado m, es computacionalmente difícil obtener un m' tal que $h(m) = h(m')$.

Tipicamente, la salida de estas funciones son de tamaños de 128, 160, 256 o 512 bits. Es importante mencionar que, a diferencia de otros algoritmos descritos en secciones anteriores, este no requiere del uso de una llave. De tal forma que cualquiera puede calcular h dada su descripción pública.

Existen diversas funciones hash, tanto criptográficas como no criptográficas. Muchas de estas no cumplen con la última carácteristica mencionada anteriormente, esto debido a ataques relacionados a la longitud del digesto resultante. En la tabla 3.1 presentamos algunos ejemplos, indicando en cuales se han encontrado colisiones y por lo tanto no pueden ser usadas en sistemas criptográficos. Actualmente los estándares del NIST, recomiendan el uso de SHA-256 en adelante [72, 73].

Las funciones hash tienen diversas aplicaciones en criptografía. A continuación describiremos las más importantes.

Algoritmo	Longitud	Notas
MD4	128	Se han encontrado colisiones
MD5	128	Se han encontrado colisiones
SHA-1	160	Se han encontrado colisiones
SHA-256	256	Sin colisiones documentadas
SHA-512	512	Sin colisiones documentadas
SHA-3	224 a 512	Sin colisiones documentadas

Tabla 3.1: Algoritmos de hash más conocidos

3.3.2. Funciones hash para integridad

Una de las aplicaciones criptográficas más conocidas de los algoritmos de *hash* es la verificación de integridad de un mensaje. Aquí estas funciones se usan para detectar si un mensaje ha sido modificado de manera no autorizada. Su funcionamiento consiste en calcular el digesto del mensaje original para despúes usar este como prueba para una posible verificación de si el mensaje ha sido modificado. A las funciones *hash* diseñadas con este objetivo se las llama Códigos de Detección de Modificaciones o MDC (del inglés *Modification Detection Codes*)

Para cumplir su objetivo la función *hash* debe ser criptográfica. Esto es tiene que cumplir con propiedades que la hagan resistente frente ataques de adversarios maliciosos cuyo objetivo es que la función no cumpla su objetivo. De tal manera que esta función debe de cumplir con las propiedades mencionadas en la Sección 3.3.1.

La manera en que esta función se usará para validar la modificación no autorizada de un mensaje se describe en la Figura 3.4. Ahí podemos ver como el emisor de un mensaje calcula el digesto de dicho mensaje, para después enviar

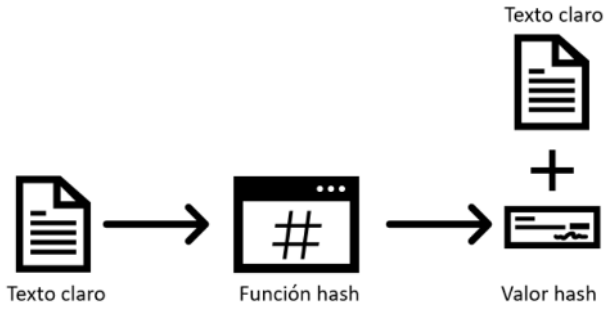


Figura 3.4: Esquema de generación de valor hash

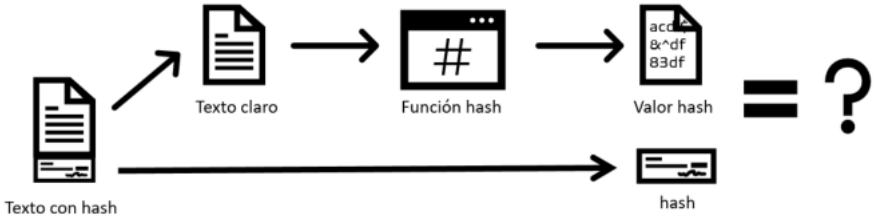


Figura 3.5: Esquema de verificación de integridad

ambos elementos en el canal. Cuando el receptor recibe estos dos elementos, los separa, re-calcula el digesto del mensaje recibido y lo compara con el hash recibido. Si estos son iguales significa que no hubo modificación. Si por el contrario, son diferentes, significa que existió modificación y usualmente el mensaje se rechaza. Esto se puede observar en la Figura 3.5.

3.3.3. Funciones hash para autenticación de mensajes

Otra aplicación que se le da a las funciones *hash* es el de autenticar el origen de un mensaje. Estas funciones se conocen como Códigos de Autenticación de Mensajes o MAC (del inglés *Message Authentication Codes*).

Los MAC se caracterizan por el utilizar una llave secreta para poder calcular la integridad del mensaje. Bajo el supuesto de que esta llave sólo es conocida por el emisor y el receptor, se consigue que el receptor verifique tanto la integridad del mensaje como la posesión de la correspondiente llave privada. Esto se realizaría de la siguiente manera.

$$\text{MAC} = h_K(M)$$

donde:

- M es un mensaje en claro de longitud arbitraria.
- h_K es la función *hash* que recibe un mensaje en claro y utiliza una llave secreta K para calcular el valor del hash MAC.

Si el valor MAC del emisor es idéntico al calculado por el receptor, entonces ser puede garantizar que:

1. El mensaje no sufrió modificaciones no autorizadas.
2. El mensaje proviene del remitente indicado en el mensaje (el dueño de la llave).
3. Si el mensaje incluye un número de secuencia, que el mensaje sigue la secuencia correcta.

3.3.4. Cadena de hash

Una cadena hash o *hash chain* está formada por la aplicación sucesiva de una función hash criptográfica a un mensaje. De tal manera que siendo h una función hash y m dicho mensaje, la aplicación sucesiva de h 4 veces sería como se muestra a continuación:

$$h(h(h(h(m))))$$

Esta la cadena hash de longitud 4 se denota como $h^4(m)$ o simplemente h_4 Si generalizamos este proceso entonces tendríamos que,

$$h^n(m) = h(...h(h(h(m))))$$

Lamport [74] sugirió el uso de cadenas de hash como un esquema de protección de contraseñas en un entorno inseguro. Por ejemplo, si un servidor requiere proporcionar autenticación a un grupo de usuarios, en lugar de almacenar las contraseñas de estos usuarios en texto plano, puede almacenar una cadena de hash de dichas contraseñas evitando el robo de estas ya sea en la transmisión o en el almacenamiento.

Por ejemplo, un servidor comienza almacenando h^{1000}(contraseña) donde la contraseña es la proporcionada por el usuario. Cuando el usuario desea autenticarse, envia al servidor h^{999}(contraseña). El servidor calcula $h(h^{999}$(contraseña)) = h^{1000}(contraseña) y verifica que coincide con la cadena de hash que tiene almacenada. En ese momento el servidor almacena h^{999}(contraseña) para ser usado la próxima vez que el usuario requiera autenticarse.

De esta manera, si un atacante es capaz de capturar h^{999}(contraseña) durante la comunicación entre el usuario y el servidor, este será incapaz de retransmitir la cadena de *hash* que el servidor requiere para la autenticación. Esto porque

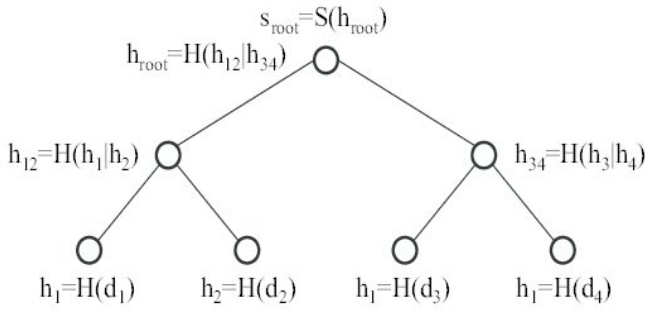


Figura 3.6: Ejemplo de un árbol de hash

el servidor esperará h^{998}(contraseña). Debido a la propiedad unidireccional de las funciones *hash* criptográficas, es computacionalmente difícil que el atacante invierta esta función y obtenga la pieza anterior de la cadena de *hash*. En este ejemplo, el usuario podría autenticarse 1000 veces antes de que se agotara la cadena de *hash*.

3.3.5. Árbol de hash

Un árbol de *hash* o árbol de Merkle es un árbol en el que cada nodo hoja contiene el *hash* criptográfico de un dato y cada nodo no hoja está contiene el *hash* criptográfico del contenido de sus nodos hijos, como se puede observar en la Figura 3.6. Los árboles de *hash* permiten una verificación eficiente y segura del contenido de grandes estructuras de datos siendo estos la generalización de las listas de *hash* y las cadenas de *hash*. Este concepto fue presentado por Ralph Merkle en [75].

Los árboles de *hash* pueden usarse para verificar cualquier tipo de datos almacenados, manejados y transferidos dentro y entre computadoras. Pueden garantizar que los bloques de datos recibidos en una red punto a punto se reciban

sin daños ni alteraciones, e incluso para comprobar que los integrantes de esta no mienten y envíen bloques falsos.

También se utilizan en algunos sistemas de archivos como IPFS, Btrfs y ZFS para contrarrestar la degradación de los datos. En el protocolo Dat, el protocolo Apache Wave [77], los sistemas de control de revisiones distribuidas como Git y Mercurial, así como también en las redes de Bitcoin y Ethereum [76]. Sistemas de bases de datos no SQL como Apache Cassandra, Riak y Dynamo tambien los usan.

3.3.6. Otras Aplicaciones

Las funciones *hash* tienen diversas aplicaciones, a continuación describimos algunas que consideramos importantes.

Construcción de esquemas de compromiso. Los esquemas de compromiso permiten que una entidad elija un valor entre un conjunto finito de posibilidades de tal forma que no pueda cambiarla. Esa entidad no tiene que revelar su elección hasta si acaso el momento final (la elección puede permanecer oculta).

Construcción de algoritmos de cifrado/descifrado. Se usa en la construcción de algunos modos de funcionamiento en los cifradores por flujo y cifradores por bloque.

Construcción de cadenas pseudoaleatorias. El modelo de oráculo aleatorio se basa en considerar que funciones hash con ciertas propiedades se comportan como funciones que escogen cadenas al azar, se usa para el estudio de la seguridad los esquemas criptográficos.

Firma digital En los esquemas de firma digital, normalmente en lugar de firmar todo el contenido se suele ser firmar solo el valor hash del mismo. Esto se explica de manera detallada en la siguiente capítulo.

Suma de verificación. Tambien conocida como *checksum*, se realiza cuando se quiere almacenar o transmitir información y protegerla de errores accidentales. Los datos originales se acompañan de valores hash obtenidos a partir de ellos de forma que puedan ser usados para verificar la integridad del dato original.

Protección de contraseñas. Para comprobar si una contraseña de autenticación es correcta no es recomendable el almancenamiento de estas contraseñas en su versión original en claro. En su lugar se almacena el valor hash de las contraseñas. Así, para verificar si una contraseña es correcta basta con aplicar la función *hash* correspondiente y verificar si el resultado coincide con el que tenemos almacenado.

3.4. Criptografía moderna en México

La criptografía de llave simétrica y asimétrica es muy utilizada en los sistemas criptográficos que se usan diariamente en millones de transacciones alrededor del mundo, en donde ha demostrado que son capaces de ofrecer la seguridad que los sistemas requieren.

La criptografía simétrica suele ser muy eficiente, pues en general, requieren tiempos de procesamientos cortos lo que permite cifrar cantidades enormes de información.

La criptografía es un área muy activa de investigación en el mundo y en México, de hecho existen grupos de investigación importantes que estudian este tema.

Capítulo 4

Criptografía moderna: firma digital

Resumen

Este capítulo presenta la firma digital como un algoritmo criptográfico para proporcionar los siguientes servicios de seguridad: autenticación, integridad y no rechazo. El capítulo describe el algoritmo RSA, sus aplicaciones en México y su futuro.

4.1. Introducción

El diccionario de la real academia española define la firma como el "rasgo o conjunto de rasgos, realizados siempre de la misma manera, que identifican a una persona y sustituyen a su nombre y apellidos para aprobar o dar autenticidad a un documento".

Historicamente la firma se remonta al origen de la humanidad, que mediante una representación gráfica de figuras humanas se identificaba, inicialmente asociada a grupos de cazadores y recolectores, pero también con su medio ambiente (flora, fauna y formas geométricas). En México, existen varias zonas arqueológicas abiertas al público con arte rupestre, es decir con manifestaciones gráficas grabadas y pintadas sobre rocas y en cuevas, destacando las ubicadas en las sierras de la península de Baja California Sur, como muestra la foto de la Fig. 4.1 tomada en el Museo de Antropología e Historia de Baja California Sur, La Paz, México.

En Computación, un archivo digital está formado de cadenas de ceros y unos los cuales pueden representar caracteres alfanuméricos, es decir un texto, o imágenes y figuras, o sonidos y música, etc.

En una computadora un archivo digital facilmente se puede crear, copiar, insertar, modificar y borrar por lo que se requiere de un mecanismo que permita detectarlo. La firma digital es un mecanismo criptográfico que permite verificar la autenticidad, integridad y no rechazo de un archivo digital.

En México, la firma digital está legislada por el decreto expedido por el Diario Oficial de la Federación (DOF) el 11 de enero de 2012 el cual establece la Ley de Firma Electrónica Avanzada y en el DOF del 21 de marzo de 2014 se publica su reglamento, es decir que la Firma Electrónica Avanzada tiene los mismos propósitos legales que la firma manuscrita.

Figura 4.1: Museo de Antropología e Historia de Baja California Sur, La Paz, México

En esta sección revisaremos brevemente la criptografía para implementar la Firma digital, sus aplicaciones y su futuro.

4.2. Criptografía

La criptografía es el estudio sistemático de las técnicas matemáticas para ofrecer los servicios de seguridad a la información, tales como: confidencialidad, integridad, autenticación, control de acceso y no rechazo.

La *confidencialidad* permite proteger la información contra revelaciones no autorizadas; la *integridad* permite detectar modificaciones de la información no autorizadas; la *autenticación* permite verificar la identidad de un usuario o de un sistema; el *control de acceso* protege el acceso a la información; y el *no rechazo* per-

mite detectar la negación de una actividad, por ejemplo que un emisor niegue haber enviado un mensaje o que el receptor niegue haberlo recibido.

Para implementar estos servicios, el Instituto Nacional de Estándares y Tecnologías (NIST) del Departamento de Comercio del Gobierno de los Estados Unidos de Norteamérica ha publicado diferentes estándares donde se especifican los algoritmos y los protocolos criptográficos para que todas las dependencias del gobierno norteamericano protejan su información.

En particular el estándar NIST FIPS 186-4 [78] especifica tres algoritmos criptográficos para implementar la firma digital, a saber: Algoritmo de Firma Digital (DSA), Algoritmo de Firma Digital RSA y Algoritmo de Firma Digital basado en Curvas Elípticas (ECDSA).

RSA es el algoritmo más ampliamente usado en el mundo por lo que a continuación detallaremos su funcionamiento.

4.2.1. Criptografía Asimétrica

In 1976 [79] Whitfield Diffie y Martin Hellman (ganadores del Premio Turing 2015, equivalente al premio Nobel en Computación) fueron los primeros en proponer públicamente los conceptos de la Criptografía de Llave Pública (PKC, *Public Key Cryptography*) o Asimétrica y un protocolo de intercambio de llaves.

Como se describió en la Sección 3.2, en PKC, cada usuario tiene dos llaves, una privada (que mantiene en secreto) y una pública conocida por el resto de los usuarios en el sistema. Ambas llaves están relacionadas matemáticamente en tal forma que conociendo la llave pública y el algoritmo de cifrado es computacionalmente imposible obtener la llave privada o secreta, pero es relativamente fácil

cifrar y descifrar usando una de las dos llaves; cualquiera de las dos llaves se puede usar para cifrar mientras que la otra se usa para descifrar.

Computacionalmente imposible significa que un atacante usando todos los recursos computacionales a su alcance no puede calcular la llave privada. Aquí un atacante puede ser una persona o un grupo de personas o gobiernos con una gran infraestructura computacional.

La PKC está basada en problemas que la Computación no puede resolver de manera eficiente, es decir problemas cuyos algoritmos de solución se ejecutarán en tiempos que no se pueden acotar en un tiempo polinomial conforme el tamaño de los datos de entrada se incrementa. A esta clase de problemas se les llama NP (non-polynomial) y fueron previamente descritos en la Sección 3.2.

En 1977 [80] Ron Rivest, Adi Shamir y Len Adleman (ganadores del premio Turing 2002), mostrados en la Fig. 4.2, propusieron usar el problema de factorización de números enteros.

Figura 4.2: Ganadores del Premio Turing 2002 [81]

Bien sabemos que un número primo es aquel que sólo es divisible por la unidad y él mismo, y son difíciles de encontrar cuando son muy grandes. Por ejem-

plo, el Museo de Ciencias de la UNAM tiene una colección de números primos gigantes, el más grande descubierto en 2018 por el "Great Internet Mersenne Prime Search" (GIMPS) tiene 24,862,048 dígitos, es el número $2^{82589933} - 1$, y requiere 11 libros para imprimirlo, como se muestra en la Fig. 4.3.

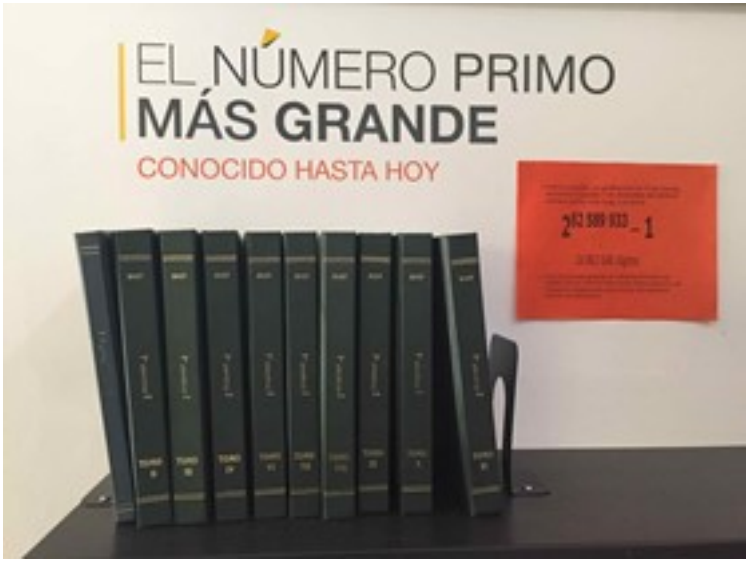


Figura 4.3: El número primo más grande

4.2.2. Firma digital RSA

Basados en el problema de la factorización de números enteros muy grandes, Rivest, Shamir y Adleman (RSA) propusieron el siguiente algoritmo 4.1:

Algoritmo 1 Generación de llaves

1: Selecciona dos números primos muy grandes: p, q

2: Calcula $n = pq$

3: Calcula $\phi(n) = (p-1)(q-1)$

4: Selecciona aleatoriamente la llave pública e, tal que: $1 < e < \phi(n)$, $mcd(e, \phi(n)) = 1$ (donde mcd es el máximo común divisor)

5: Resuelve la siguiente ecuación para encontrar la llave privada: $ed \equiv 1$ mód $\phi(n)$ donde $0 \leq d \leq n$ y mód es la operación módulo

6: Publica la llave pública: $PK = (n, e)$

7: Mantén en secreto la llave privada: $SK = (d, p, q)$

Algoritmo 2 Generación de la Firma RSA

`Entrada:` Llave pública RSA (n, e), Llave privada RSA (d), mensaje (m)

`Salida:` Firma (s)

1: Calcula $h = Hash(m)$; donde Hash es una función Hash

2: Calcula $s \equiv h^d$ mód n;

3: Regresa (s);

Algoritmo 3 Verificación de la Firma RSA

`Entrada:` Llave pública RSA (n, e), mensaje (m), firma (s).

`Salida:` Acepta o rechaza la firma.

1: Calcula $h = Hash(m)$;

2: Calcula $h' \equiv s^e$ mód n;

3: Si $h = h'$ entonces Regresa ("Firma aceptada"); de otra forma Regresa ("Firma rechazada");

4.2.3. Las matemáticas de RSA

Como se puede observar en la sección anterior, el algoritmo RSA es muy simple y de ahí su elegancia; sin embargo, tiene una base matemática muy poderosa como se describe brevemente a continuación.

1. $\phi(n)$ es la función ϕ de Euler y calcula el número de primos relativos entre $1 \leq a \leq n$.

2. mcd(a, b) es el máximo común divisor de a y b.

3. mcd(a, b) = 1 significa que a y b son primos relativos, es decir que el único divisor común es la unidad.

4. mód es la operación módulo definida en los número enteros como: $a \equiv b$ mód $n \iff a = b + nk$ con k entero (se lee a es congruente con b módulo n).

Nota que la operación se realiza en el conjunto de número enteros ($-\infty, \ldots, -3, -2, -1, 0, 1, 2, 3, \ldots, \infty$) negativos y positivos.

Ejemplo 1: $a \equiv 7$ mód $5 \iff a = 7 + 5k$, entonces a puede tomar los valores de 12, 17, 22, 27, 32, 37 cuando $k = 1, 2, 3, 4, 5$ y 6, respectivamente. Además nota que a y b tienen el mismo residuo al dividirlos por n.

5. Hash es una función como las ya descritas en la Seccion 3.3 que aquí llamaremos *resumen*.

6. $s \equiv h^d$ mód n: es la función de exponenciación módulo n.

Ejemplo 2: $s \equiv 7^3$ mód $8 \equiv 343$ mód $8 = 7$

Ejemplo 3: $s \equiv 2^{2020}$ mód 789, si usamos la calculadora para calcular 2^{2020} marcará infinito o error, es decir la memoria de la calculadora se desbordó y no pudo realizar el cálculo. Una forma eficiente de hacerlo es calculando las potencias de 2 módulo 789, como sigue:

$$
\begin{aligned}
2^2 &\equiv 4 \text{ mód } 789 = 4 \\
2^4 &\equiv 4^2 \text{ mód } 789 = 16 \\
2^8 &\equiv 16^2 \text{ mód } 789 = 256 \\
2^{16} &\equiv 256^2 \text{ mód } 789 \equiv 65536 \text{ mód } 789 = 49 \\
2^{32} &\equiv 49^2 \text{ mód } 789 \equiv 2401 \text{ mód } 789 = 34 \\
2^{64} &\equiv 34^2 \text{ mód } 789 \equiv 1156 \text{ mód } 789 = 367 \\
2^{128} &\equiv 367^2 \text{ mód } 789 \equiv 134689 \text{ mód } 789 = 559 \\
2^{256} &\equiv 559^2 \text{ mód } 789 \equiv 312481 \text{ mód } 789 = 37 \\
2^{512} &\equiv 37^2 \text{ mód } 789 \equiv 1369 \text{ mód } 789 = 580 \\
2^{1024} &\equiv 580^2 \text{ mód } 789 \equiv 336400 \text{ mód } 789 = 286
\end{aligned}
$$

Dado que el número decimal 2020 = 1024 + 512 + 256 + 128 + 64 + 16

El número 2020 se puede representar en un número binario, es decir, en una cadena de o's y 1's, de la siguiente manera: $2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^4$, es decir 11111010000.

De esta forma, 2^{2020} mód 789 $\equiv 286 \times 580 \times 37 \times 559 \times 367 \times 49 \equiv$ 61697803487320 mód789 $\equiv$ 535 mód 789.

Notemos que la función de exponenciación módulo n, $s \equiv h^d$ mód n, realiza el cifrado del mensaje h usando la llave privada d y es relativamente fácil de

calcular conociendo d y n. Por otro lado, la verificación de la firma es el descifrado de la firma, s^e mód n, usando la llave pública e.

4.2.4. Fortaleza de RSA

La fortaleza o nivel de seguridad de un algoritmo criptográfico se define como el número de operaciones necesarias para romperlo y se mide en bits. Ejemplo, el nivel de seguridad de 128 bits, requerirá 2^{128} operaciones para romper el algoritmo, es decir 3.4×10^{38} operaciones.

En el algoritmo RSA un atacante debe resolver los siguientes problemas para calcular la llave privada:

1. Encontrar los números primos p y q, dado que conoce n.
2. Resolver la ecuación $d \equiv e^{-1}$ mód $\phi(n)$, dado que conoce e.

En marzo de 1991, la entonces compañía *RSA Laboratories* propuso un reto a nivel mundial para factorizar 54 números semi-primos, es decir números con sólo dos factores primos, con el fin de promover la investigación en la teoría de números y para mostrar la dificultad de factorizar números primos muy grandes.

El reto terminó en 2007 con la factorización de 12 números primos; sin embargo, el reto ha continuado hasta hoy con la factorización de 11 números más. De éstos, el número primo más grande tiene una longitud de 829 bits (250 dígitos), denominado RSA-250 [82]. En 2009 los autores de la factorización de RSA-768 concluyeron que RSA-1024 podría ser factorizado en 10 años, cosa que no ocurrió. El NIST previniendo futuros ataques recomendó desde 2013 no usar más el módulo 1024 y en su lugar usar el módulo 2048.

La Tabla 4.1 resume el nivel de seguridad de RSA, donde k es la longitud del producto de los dos números primos p y q, llamado el módulo de n. Nota

Tabla 4.1: Nivel de Seguridad de RSA [87]

Algoritmo	Nivel de Seguridad (bits)	Parámetros (en bits)	Longitud de la firma (en bits)	Suposición de seguridad
RSA	80	$k = 1024$	1024	Problema de la factorización de números enteros
	112	$k = 2048$	2048	
	128	$k = 3072$	3072	
	192*	$k = 7680$	7680	
	256*	$k = 15360$	15360	

que los niveles señalados con (*) no están estandarizados en el estándar NIST $186 - 4$.

4.3. Infraestructura de clave pública

Como se mencionó en la Sección 4.2, cuando un usuario genera un par de claves, el o ella publica su llave pública al resto de los usuarios del sistema y mantiene en secreto su llave privada. Para publicar las llaves públicas es necesario desplegar una Infraestructura de Llave Pública (PKI) dónde existe una Autoridad Certificadora (CA) encargada de generar, firmar digitalmente y distribuir los Certificados de los usuarios. Un certificado contiene la llave pública de un usuario y la firma digital de la CA. De esta forma, cualquier usuario puede obtener y verificar la llave pública de otro usuario verificando la firma de la CA que emitió el certificado.

Figura 4.4: Nombre del propietario del Certificado

La CA es una entidad de confianza que inicialmente firma digitalmente su certificado, conocida como CA raíz, y puede a su vez generar certificados a otras autoridades certificadoras.

El estándar X.509 [84] define los formatos del certificado y de la lista de revocación de los certificados, ya que éstos tienen un periodo de vida y cuando expiran es necesario almacenarlos para poder verificar las firmas realizadas en ese periodo, como se describe más adelante.

4.3.1. Nombre del usuario

La Fig. 4.4 muestra un certificado de prueba generado por el Sistema de Administración Tributaria (SAT) de la Secretaría de Hacienda y Crédito Público (SHCP) de México. En ésta se pueden observar los campos contenidos en el Certificado del SAT, incluidos el nombre, dirección, país, identificador único y correo electrónico.

Figura 4.5: Identificador de la CA

4.3.2. Identificador de la CA

La Fig. 4.5 muestra los campos del Certificado del SAT que contienen la información de la CA que emitió y firmó el Certificado. Se puede observar:

(a) El Banco de México es la CA raíz que emite el Certificado al SAT

(b) El Certificado tiene un número de serie

(c) Indica la versión del estándar X.509

(d) Indica el algoritmo RSA y la función *Hash* (en este caso MD5) con los que se generó la firma

(e) El intervalo de tiempo durante el cual el Certificado es válido (no antes, no después)

4.3.3. Información de la clave pública

La Fig. 4.6 muestra los campos del Certificado del SAT que contienen la información de la llave pública.

Observe lo siguiente:

(a) Contiene información del algoritmo criptográfico empleado para generar la llave pública, en este caso RSA.

(b) Contiene la llave pública del usuario, $PK = (n, e)$, donde:

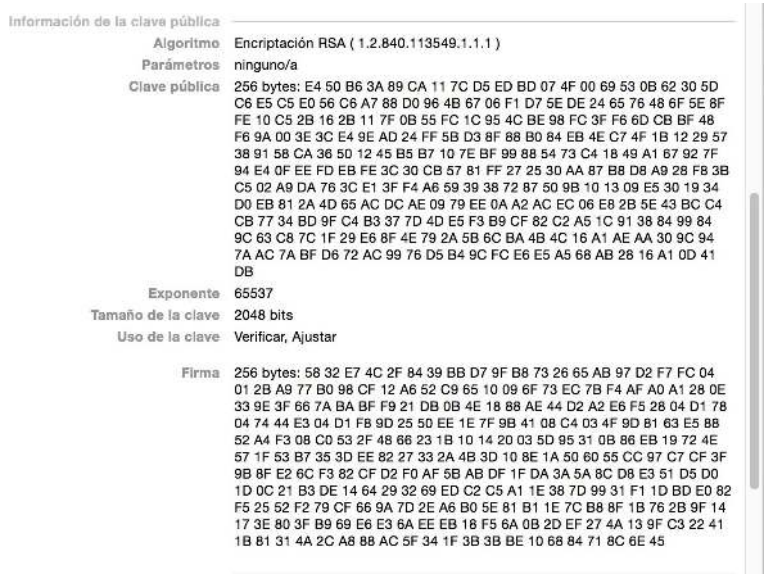
(i) n es el producto de los dos números primos (p, q) de longitud 256 bytes, es decir 2048 bits, en representación hexadecimal. En ésta, se usan 4 bits para representar a los números decimales del 0 al 15, de la siguiente manera: 0, 1, 2, . . . , 9, *A, B, C, D, E* y *F*; donde la letra *A* representa al número 10, *B* al 11, . . . y *F* al 15.

(ii) e es el exponente igual a 65537

(c) La firma de la CA de longitud 256 bytes en representación hexadecimal

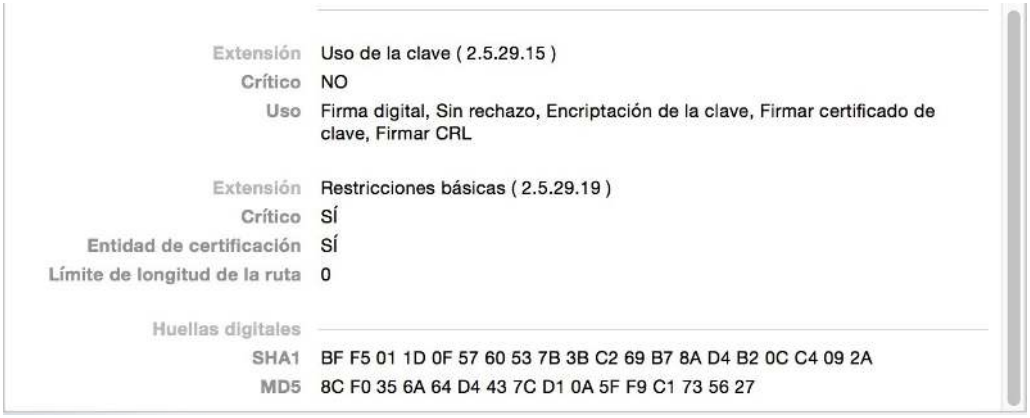
4.3.4. Resumen del Certificado

Finalmente el Certificado contiene campos para indicar: extensiones, uso y un resumen (huella digital) producido por una función Hash (en este caso las funciones SHA-1 y MD5), como se muestra en la Fig. 4.7, con el objetivo de detectar posibles modificaciones al certificado.



Información de la clave pública	
Algoritmo	Encriptación RSA (1.2.840.113549.1.1.1)
Parámetros	ninguno/a
Clave pública	256 bytes: E4 50 B6 3A 89 CA 11 7C D5 ED BD 07 4F 00 69 53 0B 62 30 5D C6 E5 C5 E0 56 C6 A7 88 D0 96 4B 67 06 F1 D7 5E DE 24 65 76 48 6F 5E 8F FE 10 C5 2B 16 2B 11 7F 0B 55 FC 1C 95 4C BE 98 FC 3F F6 6D CB BF 48 F6 9A 00 3E 3C E4 9E AD 24 FF 5B D3 8F 88 B0 84 EB 4E C7 4F 1B 12 29 57 38 91 58 CA 36 50 12 45 B5 B7 10 7E BF 99 88 54 73 C4 1B 49 A1 67 92 7F 94 E4 0F EE FD EB FE 3C 30 CB 57 81 FF 27 25 30 AA 87 B8 D8 A9 28 F8 3B C5 02 A9 DA 76 3C E1 3F F4 A6 59 39 38 72 87 50 9B 10 13 09 E5 30 19 34 D0 EB 81 2A 4D 65 AC DC AE 09 79 EE 0A A2 AC EC 06 E8 2B 5E 43 BC C4 CB 77 34 BD 9F C4 B3 37 7D 4D E5 F3 B9 CF 82 C2 A5 1C 91 38 84 99 84 9C 63 C8 7C 1F 29 E6 8F 4E 79 2A 5B 6C BA 4B 4C 16 A1 AE AA 30 9C 94 7A AC 7A BF D6 72 AC 99 76 D5 B4 9C FC E6 E5 A5 68 AB 28 16 A1 0D 41 DB
Exponente	65537
Tamaño de la clave	2048 bits
Uso de la clave	Verificar, Ajustar
Firma	256 bytes: 58 32 E7 4C 2F 84 39 BB D7 9F B8 73 26 65 AB 97 D2 F7 FC 04 01 2B A9 77 B0 98 CF 12 A6 52 C9 65 10 09 6F 73 EC 7B F4 AF A0 A1 28 0E 33 9E 3F 66 7A BA BF F9 21 DB 0B 4E 18 88 AE 44 D2 A2 E6 F5 28 04 D1 78 04 74 44 E3 04 D1 F8 9D 25 50 EE 1E 7F 9B 41 08 C4 03 4F 9D 81 63 E5 88 52 A4 F3 08 C0 53 2F 48 66 23 1B 10 14 20 03 5D 95 31 0B 86 EB 19 72 4E 57 1F 53 B7 35 3D EE 82 27 33 2A 4B 3D 10 8E 1A 50 60 55 CC 97 C7 CF 3F 9B 8F E2 6C F3 82 CF D2 F0 AF 5B AB DF 1F DA 3A 5A 8C D8 E3 51 D5 D0 1D 0C 21 B3 DE 14 64 29 32 69 ED C2 C5 A1 1E 38 7D 99 31 F1 1D BD E0 82 F5 25 52 F2 79 CF 66 9A 7D 2E A6 B0 5E 81 B1 1E 7C B8 8F 1B 76 2B 9F 14 17 3E 80 3F B9 69 E6 E3 6A EE EB 18 F5 6A 0B 2D EF 27 4A 13 9F C3 22 41 1B 81 31 4A 2C A8 88 AC 5F 34 1F 3B 3B BE 10 68 84 71 8C 6E 45

Figura 4.6: Llave pública del usuario y la firma de la CA



Extensión	Uso de la clave (2.5.29.15)
Crítico	NO
Uso	Firma digital, Sin rechazo, Encriptación de la clave, Firmar certificado de clave, Firmar CRL
Extensión	Restricciones básicas (2.5.29.19)
Crítico	SÍ
Entidad de certificación	SÍ
Límite de longitud de la ruta	0
Huellas digitales	
SHA1	BF F5 01 1D 0F 57 60 53 7B 3B C2 69 B7 8A D4 B2 0C C4 09 2A
MD5	8C F0 35 6A 64 D4 43 7C D1 0A 5F F9 C1 73 56 27

Figura 4.7: Resumen del Certificado

4.4. Aplicaciones de la Firma Digital en México

La firma digital tiene amplias aplicaciones porque provee los siguientes servicios de seguridad: autenticación, integridad y no rechazo.

Autenticación porque sólo el dueño de la llave privada (d) puede cifrar el resumen producido por la función Hash ($s \equiv h^d \text{ mód } n$) y no puede negar haber firmado porque él es el único poseedor de la llave privada.

Integridad porque la función Hash (discutida en la Sección 3.3 de este libro) tiene las siguientes propiedades: resistente a colisiones, resistente a la primera preimágen y resistente a la segunda preimágen.

En México el uso oficial de la Firma Digital está publicado en el Diario Oficial de la Federación mediante la expedición de la Ley de Firma Electrónica Avanzada (DOF: 11/01/2012), el Reglamento de la Ley de Firma Electrónica Avanzada (DOF: 21/03/2014) y las Disposiciones Generales de la Ley de Firma Electrónica Avanzada (DOF: 21/10/2016).

En el artículo 23 de la Ley se especifica que la Secretaría de la Función Pública, la Secretaría de Economía y el Servicio de Administración Tributaria (SAT) son consideradas como Autoridades Certificadoras para emitir Certificados Digitales. Sus atribuciones y obligaciones quedan definidas en el artículo 25, en particular la fracción V establece: garantizar la autenticidad, integridad, conservación, confidencialidad y confiabilidad de la firma electrónica avanzada, así como de los servicios relacionados con la misma.

El uso de la firma digital en México es cada vez más extendido, por ejemplo el SAT lo usa para la generación de los Comprobantes Fiscales Digitales por Internet (CFDI); la Suprema Corte de Justicia de la Nación regula el uso de la Firma Electrónica Certificada del Poder Judicial de la Nación (FIREL) y el Expediente Electrónico; el Instituto Mexicano del Seguro Social (IMSS) autoriza el uso de

Figura 4.8: Resumen del Certificado

la Firma Electrónica Avanzada, cuyo certificado digital sea emitido por el SAT, en las actuaciones, actos administrativos y en los procedimientos inherentes a los procesos para el otorgamiento de las prestaciones en dinero que prevé la Ley del Seguro Social.

Además, el Instituto del Fondo Nacional de la Vivienda para los Trabajadores (INFONAVIT) por el que se autoriza el uso de la Firma Electrónica Avanzada, cuyo certificado digital sea emitido por el SAT, en sus actuaciones como Organismo Fiscal Autónomo; así como en algunos estados de la República como Guanajuato, Sonora, Chiapas, Jalisco e Hidalgo.

En el caso del SAT, la Secretaría de Hacienda y Crédito Público publicó, en el DOF, el Anexo 20 de la resolución de misceláneas fiscales de los años 2006, 2010, 2012, 2014 y 2017. Este anexo es un estándar computacional que especifica la estructura, forma, sintaxis, formato y criptografía de las facturas producidas electrónicamente.

Los requerimientos de los Comprobantes Fiscales Digitales por Internet (CFDI) son los siguientes: infalsificable, único, protección contra modificaciones, protección contra no rechazo y almacenamiento a largo plazo. Estos requerimiento se satisfacen por medio de dos firmas digitales: una generada por el emisor del CFDI, llamada Certificado de Sello Digital, y otra por el SAT, llamada Sello Digital del SAT, como se muestra en la Fig. 4.8.

Nota que el SAT o cualquier contribuyente puede verificar el contenido del CFDI teniendo acceso a la cadena original, que también acompaña al CFDI, y a los certificados del emisor y del SAT.

4.5. Futuro de la Firma Digital

Como se describió en la Sección 4.2.1, el algoritmo RSA está basado en un problema difícil de resolver usando toda la infraestructura computacional que disponga un atacante; sin embargo, el desarrollo de la tecnología ha permitido construir las primeras computadoras cuánticas y aunque aún no tienen gran capacidad de cómputo, es decir el número de operaciones por segundo y la capacidad de almacenamiento está limitado, ya se publicaron algunos algoritmos para ejecutarlos en estas computadoras.

Como se mencionó antes, en 1994, Shor [70] publicó dos algoritmos resuelven los problemas de la factorización de números enteros y del logaritmo discreto en un tiempo polinomial en una computadora cuántica. Esto problemas son en los que están basados las firmas digitales RSA, DSA y ECDSA. Es decir, una vez que existan las grandes computadoras cuánticas todas las aplicaciones que hagan uso de estas firmas serán vulnerables a ataques criptográficos.

Por esta razón, en 2016 el NIST publicó una convocatoria para proponer algoritmos criptográficos de llave pública post-cuánticos, es decir algoritmos resistentes a ataques por computadoras cuánticas y clásicas, que serán estandarizados en el NIST 186-4.

En 2017, el NIST recibió 82 propuestas, de las cuales aceptó 69 como candidatos en el primera ronda. Después de su estudio en 2019 aceptó 26 candidatos

en la segunda ronda, de los cuales 17 son algoritmos de llave pública y 9 son algoritmos de firma digital.

En 2020, el NIST publicó los candidatos finalistas de la tercera ronda [85], los algoritmos de firma digital pos-cuánticos son los siguientes: Crystals-Dilithium, Falcon y Rainbow. Además aceptó como algoritmos alternos a GeMSS, Picnic y SPHINCS+.

Estos algoritmos de firma digital están basados en otra clase de problemas difíciles, que se analizan en el Capítulo 6 de este libro. En 2024 el NIST planea publicar las versión preliminar del nuevo estándar post-cuántico de firma digital por lo que será necesario especificar e implementar estos nuevos algoritmos en todas las aplicaciones que la usen en México.

Capítulo 5

Aplicaciones selectas de la criptografía moderna

Resumen

Existen múltiples aplicaciones que requieren de algoritmos criptográficos para brindar servicios o productos de manera segura. En este capítulo se discuten dos de ellas, la seguridad en bases de datos como servicio y la seguridad en la redes vehiculares. La primera aplicación considera el escenario en qué el dueño de la información la delega a un proveedor que no es confiable, la segunda discute cómo se puede brindar seguridad en las comunicaciones dentro de las redes vehiculares ad-hoc. Ambas aplicaciones tienen características especiales que hacen que los algoritmos criptográficos que son utilizados en otras aplicaciones más tradicionales como el correo seguro, comunicaciones de servidores sobre internet, entre otras; no sean directamente aplicables. Por lo que estas aplicaciones vanguardistas han generado nuevas propuestas en la literatura especializada.

5.1. Introducción

En la actualidad se generán grandes cantidades de información, por ejemplo, fotografías, videos, archivos de texto, entre otros. Por lo general, estos activos digitales contienen datos que se consideran sensibles, ya que si una entidad no autorizada los compromete puede usarlos para generar consecuencias negativas que van desde la denegación de servicios, hasta pérdidas económicas. Estos activos digitales son generados, transmitidos y almacenados por medio de distintas tecnologías y aplicaciones, que imponen requerimientos y restricciones sobre los algoritmos criptográficos que pueden ser utilizados. En lo particular en este capítulo se abordan dos aplicaciones: a) las bases de datos como servicio (subcontratadas) y b) las comunicaciones en redes vehiculares ad-hoc (VANETs).

5.2. Bases de datos como servicio

Las bases de datos subcontratadas (DaaS por sus siglas en inglés) son un servicio provisto bajo el paradigma del cómputo en la nube. DaaS permite delegar la administración y el almacenamiento a un tercero conocido como el proveedor del servicio.

En el pasado el almacenamiento de las bases de datos se realizaba de forma local, es decir, todos los datos residían en las computadoras del usuario o de la empresa que los generaba. Por el contrario DaaS propone aprovechar la capacidad de los servidores externos para almacenar la base de datos y permitir que el proveedor del servicio se encarge de su adminitración, posteriormente el cliente podrá consultar y recuperar datos de su interés. En la figura 5.1 se muestra la fase en la que el cliente delega la base de datos a un servidor y en la figura 5.2 el cliente plantea consultas para que sean computadas por el servidor.

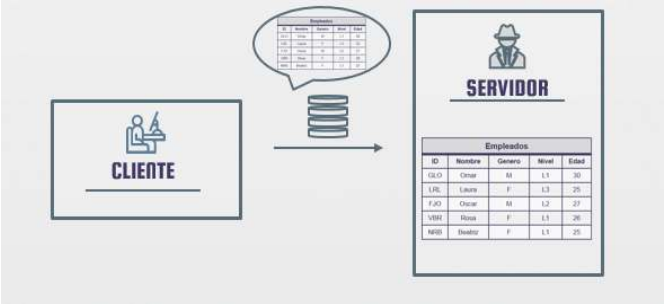


Figura 5.1: Delegación de la Base de datos

Entre las aplicaciones que permiten el almacenamiento de datos como servicio se puede mencionar: Dropbox, Google Drive; o bien el almacenamiento de bases de datos como Amazon simple Db, Microsoft Azure Cloud SQL Database, entre otras.

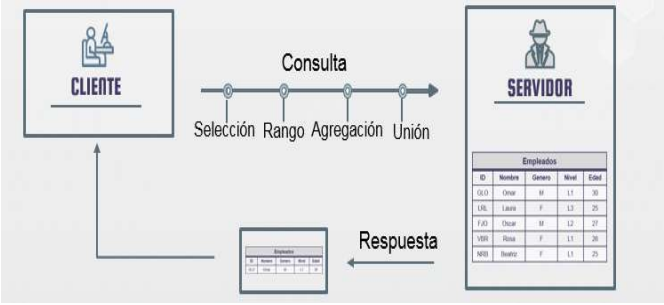


Figura 5.2: Consulta y recuperación de datos

DaaS brinda grandes ventajas para las compañías como para los usuarios finales, ya que se puede tener ahorros significativos en la administración, en los

recursos humanos especializados, licencias de software, entre otros. Sin embargo, este nuevo paradigma también conlleva riesgos de seguridad, que deben ser atendidos.

Existen tres preocupaciones principales que el dueño de la información debe considerar: a) la confidencialidad, b) la integridad y c) la disponibilidad.

En las siguientes secciones discutimos brevemente en qué consisten las inquietudes que el dueño posiblemente tendrá y que soluciones se le ha dado hasta el momento.

5.2.1. Confidencialidad

Al delegar la información a un tercero, existe la posibilidad que el proveedor de servicio exponga nuestra información o que la comparta con otras entidades de forma no autorizada. Este riesgo tiene un efecto dómino, en el caso de que la base de datos delegada pertenezca a una empresa, puesto que posiblemente se divulgaría información de sus clientes. Ante este escenario el servidor se considera no confiable.

Para proveer confidencialidad normalmente se aplica algún algoritmo de cifrado, sin embargo, el aplicar algoritmos estándar en información estructurada impide que el servidor pueda dar respuesta a las consultas de los clientes sin el conocimiento de la llave. Esto se debe a que un buen cifrador debe romper con la estructura del texto en claro (no revelar nada acerca del texto en plano que lo generó). Además, el procesar cada tipo de consulta en texto cifrado requiere de distintas propiedades del cifrador.

Considere el siguiente ejemplo: El dueño de los datos desea delegar la base de datos mostrada en la Tabla 5.1 a un proveedor DaaS, sin embargo, desconfía del

servidor y por ello preferiría cifrar los datos. Además requiere realizar consultas como:

- Q1: Seleccionar a todos los empleados de género femenino.
- Q2: Seleccionar a todos los empleados cuya edad es mayor o igual a 25 años y menor o igual de 27 años.
- Q3: Seleccionar el promedio de edad de todos los empleados.

Id	Nombre	Género	Nivel	Edad
GLO	Omar	M	L1	30
LRL	Laura	F	L3	25
FJO	Oscar	M	L2	27
VBR	Rosa	F	L1	26
MRB	Beatriz	F	L1	25

Tabla 5.1: Empleados

La consulta Q1 requiere que el cifrador aplicado sea determinista (que ante las mismas entradas se obtengan las mismas salida), de este modo el servidor podrá comparar cadenas y determinar si son iguales. La consulta Q2 requiere de un cifrador que preserve el orden, para que sea posible evaluar un rango. Finalmente la consulta Q3 requiere de un cifrador que permita realizar operaciones aritméticas en el texto cifrado en este caso sumar y posiblemente dividir, estos son conocidos como cifradores con propiedades homomorficas.

Por ello, la comunidad criptográfica está proponiendo nuevos algoritmos de cifrado que permitan que el texto cifrado filtre ciertas propiedades del texto en claro: cifradores con propiedades homomorficas [119, 117, 118] , cifradores que preservan el orden [121, 122, 123, 120], cifradores que preservan el formato

[124, 125]. Sin embargo, estos cifradores brindan un nivel de seguridad menor que por ejemplo, el tan usado AES. Por lo que deben ser estudiados con mucho detenimiento, ya que encontrar el compromiso correcto entre funcionalidad y seguridad es una tarea sumamente difícil.

5.2.2. Integridad

El problema de integridad en datos estructurados considera el escenario donde un cliente delega una base de datos a un servidor no confiable. Cuando el cliente los consulta, espera un conjunto de registros (respuesta) que satisfagan las restricciones planteadas. Como el servidor no es confiable, el cliente debe poner a prueba que las respuestas que le está enviando son correctas. Por lo tanto, debe existir un mecanismo que pueda proteger al cliente de comportamientos maliciosos, este problema es conocido en la literatura como procesamiento de consultas autenticado. El servidor debe poder proveer una prueba de que el resultado que está enviando es integro (no ha sido modificado de forma no autorizada), es completo (no está omitiendo registros que cumplen con las condiciones de la consulta o agregando registros que no satisfacen la consulta) y es fresco (corresponde a la última versión de la base de datos).

Considere el siguiente ejemplo: El dueño desea delegar la base de datos mostrada en la Tabla 5.1 a un proveedor DaaS no confiable para que almacene y administre todos los datos. Posteriormente, el dueño plantea una consulta al servidor:

- Q1: Seleccionar el nombre, género y edad de la relación Empleados donde el nivel es L1.

El servidor puede generar una respuesta correcta o incorrecta. La respuesta correcta para la consulta Q1 es la siguiente:

{(Omar, M, 30),(Rosy, F, 26),(Betty, F, 25)}

Por otro lado, una respuesta incorrecta podría se cualquiera de los siguientes casos:

- 1: {(Omar, M, 30),(Rosy, M, 26),(Betty, F, 27)}

  En esta respuesta el género de Rosy y la edad de Betty fueron modificados de manera no autorizada violentando la integridad.

- 2: {(Omar, M, 30),(Rosy, F, 26)}

  Para este otro, la respuesta es incompleta porque el registro (Betty, F, 25) no fue incluido.

- 3: Suponga que el dueño realiza una actualización de nivel L2 a L1 en la relación Empleados donde el ID es igual a FJO. Por lo que ahora la relación Empleados se muestra en la Tabla 5.2. Entonces una respuesta que no es fresca para la consulta Q1 es la siguiente:

  {(Omar, M, 30),(Rosy, F, 26),(Betty, F, 25)}

  Porque la respuesta se obtuvo de una base de datos que existió previo a la actualización (Tabla 5.1) pero que al momento en que se generó la consulta ya no es válida. Puesto que la base de datos debería ser la que se ilustra en la Tabla 5.2).

Id	Nombre	Género	Nivel	Edad
GLO	Omar	M	L1	30
LRL	Laura	F	L3	25
FJO	Oscar	M	L1	27
VBR	Rosa	F	L1	26
MRB	Beatriz	F	L1	25

Tabla 5.2: Empleados

La comunidad criptográfica se encuentra trabajando arduamente en este problema, para ello se hace uso de primitivas bien estudiadas como códigos de autenticación de mensajes [116], firmas digitales [115] y algunas estructuras de datos como árboles, bitmaps, filtros de Bloom [114, 113, 112].

5.2.3. Disponibilidad

En este servicio el cliente pregunta por algún archivo al servidor que delegó previamente, y que éste pueda enviar una prueba de que en efecto el archivo está almacenado sin necesidad de re-enviar el archivo completo. El objetivo de este servicio es garantizar que el proveedor de servicio no borre archivos del usuario de manera maliciosa o por algún error del sistema. Hay que tomar en cuenta que no toda la información almacenada es consultada continuamente por parte del cliente, por lo que el servidor podría borrar archivos y el usuario no lo descubriría hasta mucho tiempo después o posiblemente nunca.

Para ello la comunidad criptográfica está trabajando en soluciones conocidas como pruebas de conocimiento, pruebas de posesión entre otros.

5.3. Discusión

El cómputo nube ha facilitado la manera en que las personas administran e intercambian información, además ha generado un conjunto de servicios de suma utilidad para empresas y usuarios finales como es el almacenamiento o las bases de datos como servicio. Sin embargo, existen retos de seguridad que deben ser resueltos para evitar ataques de confidencialidad, integridad y disponibilidad en los datos. El hecho de que estos datos sean estructurados implica que las técnicas tradicionales de criptografía y seguridad no son directamente aplicables, por lo que la comunidad se mantiene activa buscando mejores soluciones para estos escenarios.

5.4. Seguridad en Redes Vehiculares Ad-hoc

El constante progreso tecnológico ha permitido que los automóviles esten dotados de computadoras y sensores con el objetivo de optimizar y mejorar su funcionamiento (e.g. motor, transmisión, conducción autonoma). Ahora se busca dotar a los automóviles de dispositivos que les permita comunicarse e interactuar entre ellos sin recurrir a un intermediario, creando así el paradigma de las redes vehiculares Ad-hoc (VANET por sus siglas en inglés). Las VANET ofrecen una amplia gama de aplicaciones y servicios que están estrechamente relacionados con el desarrollo de sistemas de transporte inteligente. Entre los servicios que busca impulsar este paradigma se encuentra: mejorar la eficiencia y la seguridad vial, optimizar el tráfico vehicular, la conducción cooperativa y brindar asistencia al conductor.

Este preámbulo muestra una visión futurista de las ciudades inteligentes, donde el transporte público será autónomo, los semáforos serán capaces de au-

togestionarse y los automóviles realizarán navegación cooperativa. Sin embargo, vulnerar la integridad física de los usuarios en este entorno es en un riesgo latente, ya que si una VANET es comprometida, el ataque malicioso podría ocasionar un caos vial e incluso la pérdida de vidas humanas. Por ello, es fácil concebir que estas redes necesitan de servicios de seguridad ante ataques de terceros, y así evitar escenarios trágicos. Proteger estos sistemas se convierte en un reto a causa de las características que distinguen a las redes vehiculares de otras redes tradicionales (Telefonía celular, WLAN, LAN). Cada una de estas singularidades se discuten en la siguiente sección.

5.4.1. ¿Qué características tienen las redes vehículares?

Las VANET, están integradas por vehículos e infraestructura de las calles, tal como: semáforos, señales de tránsito, entre otros, que poseen capacidades de procesamiento y de comunicación a través de redes inalámbricas, lo cuál se ilustra en la figura 5.3.

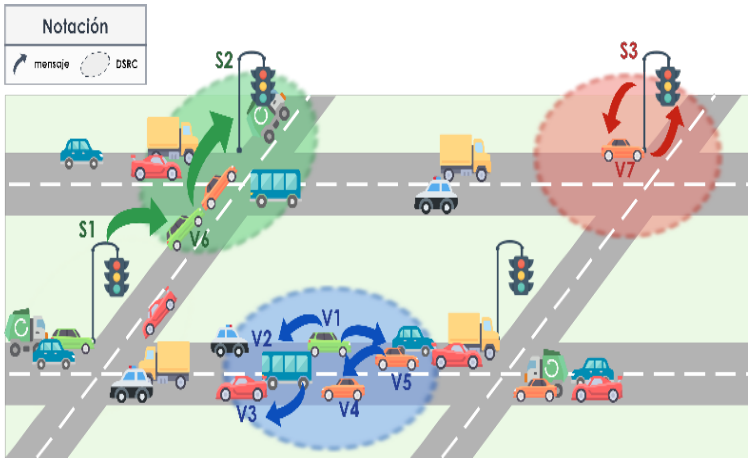


Figura 5.3: Red vehicular Ad-hoc

Entre las características que distinguen a las redes vehiculares se puede mencionar:

1. **La alta velocidad de los nodos.** Los vehículos (nodos) se encuentran en continuo movimiento y se desplazan por las calles a velocidades que van desde los 40 km/h hasta 120km/h. Esto marca una gran diferencia respecto a otras redes, donde el movimiento de los nodos es menor o nulo dentro del rango de comunicación.

2. **El rango de comunicación limitado.** Cada vehículo e infraestructura de la calle que forma parte de una VANET posee la capacidad de crear una red inalámbrica para la transmisión de información, con un rango que varía entre 90m y 500m.

3. **Cambios constantes en la topología.** Los vehículos se desplazan a distintas velocidades ocasionando una conexión y desconexión frecuente dentro de la red, esta característica hace que el tiempo para la entrega de información se vea reducido a unos segundos.

4. **Patrón de movimiento.** El movimiento de los vehículos es determinista y está restringido por la dirección de las calles, avenidas, carreteras y entronques. A diferencia de otras redes donde los nodos se pueden mover libremente y de forma aleatoria o permanecer en reposo.

5. **Sin restricción de batería.** Los dispositivos que portan los vehículos no están sujetos a restricciones de energía y, por lo tanto, son adecuados para un uso continuo.

6. **Comunicación continua entre nodos.** Los nodos obtienen información importante del entorno a través de los sensores o del conocimiento que generan otros nodos, por tanto, la comunicación constante entre los nodos es de suma importancia para el funcionamiento de las aplicaciones y servicios que ofrecen estas redes.

5.4.2. La seguridad en VANETs

Como se mencionó en la sección previa, los nodos (vehículos e infraestructura de las calles) de las VANET deben compartir información sobre su entorno. La existencia de esta comunicación permite que las aplicaciones y servicios puedan orientarse a brindar asistencia personalizada al conductor o la autoregulación del tránsito por parte de los semáforos inteligentes, entre otros. Sin embargo, este intercambio de información puede incluir datos que son sensibles y exponerlos a ciberataques, por ejemplo: considere el escenario donde el tráfico vehicular es censado continuamente por semáforos inteligentes, y la información es transportada por los vehículos hacia cada uno de los semáforos. Entonces, a partir de la información que se transmite en la red se podría: a) conocer las calles más y menos transitadas en un determinado horario, b) rastrear un vehículo en particular, c) conocer rutas y horarios de los conductores.

La consecuencias más graves no son la obtención de información o la generación de conocimiento, sino lo que se puede hacer con ella, por ejemplo: conocer los lugares más concurridos podría dar pauta a que un terrorista coloque un cochebomba en un lugar con mayor impacto, o por el contrario, los asaltos y secuestros podrían incrementar en las calles menos transitadas. Además, si no existe un canal de comunicación seguro entre las entidades que integran una VANET, un tercero podría interceptar la comunicación y modificar los datos sin el

consentimiento de las vehículos o semáforos con el propósito de alterar su funcionamiento. Cada uno de estos ejemplos muestran que al igual que en otros entornos, las aplicaciones requieren los servicios de seguridad de las aplicaciones tradicionales como: confidencialidad, integridad y autenticación, aunado al anonimato (se requiere proteger la identidad del conductor). Si bien, cada uno de estos servicios se garantizan en las redes tradicionales (excepción de anonimato) como suele ser la telefonía celular, WLAN, etc., en las redes vehiculares es un reto poder proporcionarlos, ya que las características que distinguen las VANET de otro tipo de redes como el cambio constante en la topología; dificultan la implementación de técnicas criptográficas (de seguridad) que tradicionalmente son aplicadas para la comunicación. Lo cual representa un gran desafío, ya que se requieren protocolos de comunicación capaces de transmitir un mensaje de manera rápida y segura. Por ejemplo, para evitar que entidades maliciosas analicen el contenido de los mensajes, lo que procede es cifrar los mensajes. Sin embargo, se vuelve difícil elegir cualquiera de las soluciones conocidas en la literatura: a) cifrado simétrico y b) cifrado asimétrico. La criptografía de llave simétrica requiere establecer una llave en común entre cualquiera dos entidades que se quieran comunicar. Esta característica lo hace inviable en una VANET ya que no hay conocimiento a priori de con quien se debe comunicar un automóvil. Por otra parte, la criptografía de llave asimétrica es difícil de aplicar, ya que las VANETs son redes ad-hoc, por lo que no se puede suponer la existencia de una entidad de confianza, quien sería la encargada de expedir los certificados.

5.4.3. Discusión

Las VANETs permiten una serie de aplicaciones nuevas y futuristas, sin embargo, abren una puerta a múltiples ataques cibernéticos. Además, las solucio-

nes que tradicionalmente se aplican para brindar servicios de seguridad, no son directamente aplicables en el ámbito de las VANETs debido a las características que las distinguen. Lo anterior, abre un nicho de oportunidad para generar nuevos esquemas criptográficos que permitan brindar soluciones en este contexto y que sean congruentes con las características de las VANET.

Capítulo 6

Criptografía post-cuántica

Resumen La posible llegada de computadoras cuánticas de gran envergadura permitiría aplicar el algoritmo de Shor en contra de la seguridad ofrecida actualmente por la criptografía de llave pública utilizada cotidianamente en aplicaciones informáticas masivas, tales como e-gobierno, e-comercio y e-financiero, entre otras. Dada nuestra fuerte dependencia en estas tecnologías, las consecuencias financieras serían devastadoras. Es por ello que desde hace aproximadamente una década, la comunidad criptográfica está empeñada en producir sistemas criptográficos capaces de resistir ataques lanzados desde artefactos cuánticos de gran capacidad. A la sub-disciplina surgida de estos estudios se le conoce como Criptografía Post-Cuántica. En este capítulo damos una descripción general del estado del arte actual en esta sub-disciplina.

6.1. Breve introducción a la criptografía post-cuántica

En 1982, Richard Feynman propuso el concepto de *cómputo cuántico*, en el cual planteó utilizar dos de los principios físicos fundamentales de la mecánica cuántica: la *superposición* y el *entrelazado* de estados en partículas elementales para la manipulación de datos lógicos organizados en lo que después serían llamados *qubits* (apócope del término inglés *"quantum bit"*). Típicamente, se describe un qubit como un estado cuántico que puede ser modelado en un espacio vectorial de dos dimensiones definido sobre los números complejos, en donde la base canónica del espacio $Ha_1 = C^2$, está conformada por los vectores unitarios $e_0 = [1\,0]^T$ y $e_1 = [0\,1]^T$. Si se escogen dos números complejos $z_0, z_1 \in \mathbb{C}$ tales que $|z_0^2| + |z_1^2| = 1$, entonces $z_0 e_0 + z_1 e_1$ es un estado que representa a un qubit.

Un qubit se asemeja a un bit de las computadoras clásicas en el sentido de que ambas unidades de memoria tienen dos posibles valores de salida (normalmente 0 o 1). Sin embargo, a diferencia de un bit que en un momento dado sólo puede tomar uno de esos dos valores, el valor de un qubit puede ser 0, 1 o una superposición de ambos estados lógicos. Cabe señalar que aunque han pasado ya 30 años desde su propuesta inicial, el estado de la tecnología en computadoras cuánticas está aún en sus fases iniciales, si bien es cierto que en los últimos años se han logrado resultados espectaculares que permiten alimentar la esperanza de que a mediano plazo será posible construir un ordenador cuántico con capacidades de cómputo superiores a los dispositivos clásicos disponibles hoy en día.[1]

[1] La pregunta de si será o no posible construir una computadora cuántica en un futuro próximo es objeto de un muy intenso debate, el cual es *especialmente* álgido entre los físicos teóricos

Partiendo del supuesto que los ordenadores cuánticos son una realidad y que están disponibles para su uso académico y comercial, científicos computacionales han propuesto de manera teórica una serie de algoritmos que están especialmente diseñados para tomar ventaja de las características *sui generis* que tales dispositivos electrónicos ofrecerían. Por ejemplo, se han propuesto protocolos de criptografía cuántica que permiten que dos partes, digamos *Alicia* y *Betito*, se comuniquen de manera segura sin que exista la posibilidad que un oponente, digamos *Eva*, sea capaz de alterar y/o espiar la comunicación de los datos que hayan sido intercambiados entre ellos, pues tal intromisión supondría la violación de principios físicos fundamentales que al día de hoy, son ampliamente aceptados por la comunidad científica. De esa manera, los protocolos criptográficos cuánticos prometen ofrecer lo que nunca lograron cumplir del todo los protocolos criptográficos clásicos, esto es, una comunicación segura y confidencial en la cual sea *imposible* que ningún oponente pueda modificar o discernir los datos intercambiados por las partes legítimamente involucradas en una transacción electrónica.[2]

Acaso más sorprendentemente, Peter Shor propuso en 1992 en [70] un insólito y celebrado algoritmo que de ser ejecutado en una computadora cuántica poderosa, permitiría resolver el problema de factorización entera en un tiempo con complejidad polinomial aleatoria al tamaño del número entero analizado.

---

pertenecientes a la comunidad de mecánica cuántica, una de las disciplinas de ciencia básica más puras.

[2] Sin embargo en un trabajo publicado en la prestigiosa revista *Nature photonics* por un equipo de científicos noruegos junto con un estudiante potosino [98, 101], se logró romper un protocolo cuántico comercial. Tras iluminar un pequeño laser en el detector de Betito, los autores de[98] lograron deshabilitar el detector, interceptar los bits cuánticos de la llave y convertirlo a un bit clásico que coincidía con el valor que había sido enviado previamente por Alicia.

Más aún, el algoritmo de Shor puede ser fácilmente modificado para resolver eficientemente un problema más genérico conocido como el *Problema del Subgrupo Escondido (PSE)*.[3]

La resolución del PSE en tiempo polinomial implica que los criptosistemas de firma digital empleados en la actualidad como piedra fundamental del comercio electrónico, transacciones bancarias electrónicas, e-gobierno, etc., entre los que se cuentan RSA, DSA, ECDSA y la criptografía basada en emparejamientos bilineales, entre otros, estarían todos "*rotos*" pues sus garantías de seguridad basadas en la intratabilidad computacional de la factorización de enteros y del problema del logaritmo discreto, no se sostendrían más.

Sin embargo, desde el bando de los criptógrafos no todo está perdido, pues el algoritmo de Shor *no* permite romper todos los esquemas criptográficos clásicos existentes, y a pesar de que existen otras propuestas de procedimientos cuánticos (notablemente las que utilizan el algoritmo de Grover [96]), que sí se pueden aplicar en algunos de los criptosistemas clásicos que sobrevivirían a los ataques al PSE, tales propuestas exhiben complejidad sub-exponencial, por lo que los criptosistemas atacados todavía se consideran seguros. Resulta especialmente notable, por lo inesperado, que la gran mayoría de algoritmos criptográficos simétricos (también conocidos como criptosistemas de llave secreta), sean capaces de resistir todos los ataques cuánticos desarrollados hasta ahora.

En general, se define como sistemas criptográficos **post-cuánticos** a los esquemas que están fundamentados en la intratabilidad computacional de pro-

[3] Sea $\mathbb{G}$ un grupo con $\mathbb{H} \preceq \mathbb{G}$, y sea f una función en $\mathbb{G}$. Se dice que f separa clases colaterales de $\mathbb{H}$ si $f(u) = f(v) \Leftrightarrow u\mathbb{H} = v\mathbb{H}, \forall u, v \in \mathbb{G}$. Sea $\mathcal{A}$ un oráculo que computa una función capaz de separar clases colaterales de un mismo subgrupo $\mathbb{H} \subset G$. El Problema del Subgrupo Escondido (PSE), consiste en usar la información obtenida a través de $\mathcal{A}$ para encontrar un conjunto generador de $\mathbb{H}$.

blemas difíciles para los cuales el algoritmo de Shor no se puede aplicar de manera efectiva. Algunos de los sistemas criptográficos post-cuánticos más estudiados en la actualidad garantizan su seguridad en la dificultad de resolver los siguientes problemas difíciles: determinación de los síndromes correctos en una palabra protegida con códigos de corrección de error que ha sido alterada adrede con ruido gausiano, resolución de sistemas de ecuaciones multivariables cuadráticos, búsqueda del vector más corto y del vector más cercano en retículas definidas sobre espacios vectoriales discretos, determinación de curvas isógenas definidas sobre curvas elípticas supersingulares, búsqueda de pre-imágenes en funciones picadillo, determinación de la llave secreta utilizada en un cifrador por bloques, resolución del problema de la palabra en grupos no Abelianos, hallazgo de núcleos permutados de matrices y perceptrones de matrices, etc.

Para el análisis de las bondades de un determinado criptosistema post-cuántico resulta indispensable recurrir a la teoría de la complejidad algorítmica y a su clasificación de problemas computaciones en las clases *P, NP, PSPACE* y *BQP*, las cuales pueden ser informalmente definidas como sigue.

Se define a las clases *P* y *NP*, como el conjunto de todos los problemas computacionales que pueden ser resueltos en tiempo determinístico polinomial y en tiempo no determinístico polinomial, respectivamente. La clase *PSPACE* engloba los problemas que se computan con complejidad *espacial* polinomial, mientras que la clase *BQP* reune al conjunto de problemas que pueden resolverse con una complejidad polinomial en el número de compuertas cuánticas pertenecientes a un ordenador cuántico. Se sabe que $P \subseteq NP \subseteq PSPACE$ y también que $P \subseteq BQP \subseteq PSPACE$, pero si ignora casi todo lo demás, incluyendo la que se considera la pregunta más importante de las ciencias computacionales, si acaso $P = NP$.

Desde el punto de vista de la criptografía post-cuántica la pregunta de si $BQP \subseteq NP$ es verdadera o falsa, es quizás la más relevante de todas. Si efectivamente la conjetura $BQP \subseteq NP$ es cierta, entonces habría que diseñar criptosistemas cuya seguridad descanse en problemas difíciles de la clase NP. Por el contrario, si $NP \subset BQP$, entonces la única alternativa que se vislumbra es la de definir criptosistemas cuya seguridad descanse en problemas indecidibles [91].

Algunos de los problemas de investigación en ciencia básica asociados a la criptografía post-cuántica que se consideran problemas abiertos incluyen el estudio, análisis y desarrollo de [93]: nuevos y mejores algoritmos que permitan resistir de mejor manera los ataques cuánticos sin que la complejidad de computar cifras y su descifrado, y sobretodo, firmas digitales y su verificación, crezcan desaforadamente; cuantificación de la complejidad computacional clásica y *cuántica* asociada a los problemas difíciles en los que basan su seguridad los criptosistemas cuánticos propuestos y tratar de refutar/confirmar las conjeturas de complejidad computacional enunciadas en el párrafo anterior.

6.2. Antecedentes

6.2.1. Criptografía moderna

Como se ha revisado en los capítulos previos, la criptografía tiene como principal tema de estudio el problema de cómo establecer un intercambio de información seguro a través del uso de un canal de comunicación que no lo es. En la actualidad, esta área de investigación se ha convertido en una de las disciplinas más aplicadas de las ciencias de la computación. Entre los servicios prácticos más importantes que están directamente relacionados a la criptografía podemos mencionar: cifrado de datos, dinero digital, firma-verificación digital, eleccio-

nes digitales, autentificación de usuarios y de sistemas en la red, distribución de datos y tarjetas inteligentes (*smart cards*), integridad de datos, servicios de anonimato y privacidad y muchos más. En aplicaciones criptográficas, la eficiencia y el nivel de seguridad (discreción) de las implementaciones son dos objetivos naturales, pero al mismo tiempo contradictorios que el diseñador debe intentar balancear.

En un principio, todos los sistemas criptográficos propuestos fueron diseñados con una clave o llave secreta, necesaria para realizar tanto el proceso de cifrado, como el de descifrado. En estos esquemas, conocidos como criptosistemas de clave secreta, se supone que cada una de las partes legítimamente involucradas en la comunicación son las únicas entidades que tienen conocimiento de la clave secreta. Los sistemas de clave secreta desarrollan esquemas simétricos de cifrado/descifrado, los cuales contrastan con los métodos usados por la criptografía de clave pública, que fuera propuesta por primera vez en el trabajo que Diffie y Hellman publicaron en 1976 [79], y que fuera puesta en práctica muy poco tiempo después, cuando en 1978 Rivest, Shamir y Adleman propusieron el criptosistema RSA [80].

En el cifrado con llave pública o asimétrica existen dos llaves distintas: una para cifrar y otra para descifrar. La llave para cifrar es conocida públicamente y, por ende, se denomina llave pública. La llave para descifrar sólo es conocida por el receptor del mensaje, por lo que se le denomina llave privada. La ventaja de estos sistemas criptográficos es que la denominada llave pública puede ser usada por cualquier persona para cifrar mensajes bajo la premisa que sólo quien posea la llave privada podrá descifrar dichos mensajes. Aunque los métodos de llave pública son muy poderosos, tienden a tener un costo computacional elevado.

La seguridad de los criptosistemas que están hoy en uso, ya sean de clave pública o no, subyace en la complejidad computacional de un problema matemático difícil asociado con cada esquema criptográfico. Ejemplos ya mencionados anteriormente son: La factorización de números gigantes, el cómputo de logaritmos discretos también de números gigantes, etc.

Se cree, sin que exista una certidumbre absoluta, que estos problemas son extremadamente difíciles de resolver. Como mencionamos en el Capítulo 3, la seguridad de RSA descansa en la dificultad de resolver una forma especial del problema de factorización entera, mientras que la seguridad asociada a la criptografía de curvas elípticas subyace en el problema de cómputo del logaritmo discreto en este grupo, un problema matemático extremadamente complicado de resolver.

6.2.2. Criptografía post-cuántica

Los sistemas criptográficos tradicionalmente utilizados en la industria basan su seguridad en el supuesto de la intratabilidad computacional de los siguientes problemas aritméticos: factorización de números enteros, y el problema del logaritmo discreto. En el caso de la factorización de enteros, el sistema más emblemático es RSA, mientras que para el logaritmo discreto, son los protocolos basados en el esquema Diffie-Hellman, y en las variantes bilineares: criptografía de curvas elípticas y la criptografía basada en emparejamientos.

Estos supuestos pueden ser traducidos al *problema del subgrupo escondido*. El problema del subgrupo escondido consiste en escoger un subgrupo lo suficientemente grande para que una búsqueda lineal dentro del mismo sea muy costosa respecto al beneficio obtenido.

El algoritmo cuántico de Shor (Algoritmo 4) permite resolver en tiempo polinomial aleatorio ciertos esquemas del problema de subgrupo escondido para grupos finitos Abelianos. En particular, este algoritmo puede romper los esquemas de RSA, DSA y ECDSA[4] utilizando una computadora cuántica en $O((\log N)^3)$, donde N es el tamaño en bits del número a factorizar, mientras que los esquemas más poderosos en computación clásica requieren tiempo subexponencial en el mejor de los casos. Ataques similares utilizando computación cuántica aplican a sistemas criptográficos basados en el mismo problema computacional.

Algoritmo 4 Algoritmo de Shor propuesto en [70]

Entrada: $N \in \mathbb{N}$ no primo

Salida: un factor de N

1: Escoja un x aleatorio entre el rango $[2..N]$
2: Si el MCD$(x, N) \neq 1$ terminar
3: Encontrar el orden r de x mód $N (x^r \equiv 1 \text{ mód } N)$
4: **si** r es par y $x^{r/2} \neq \pm 1$ mód N **entonces**
5: MCD$(x^{r/2} + 1, N)$ es un factor no trivial de N
6: **si no**
7: intentar con otra x
8: **fin si**

En general, se define como sistemas criptográficos **post-cuánticos** a los esquemas que están fundamentados en la intratabilidad computacional de problemas difíciles para los cuales el algoritmo de Shor no se puede aplicar de ma-

[4] los más utilizados en la actualidad y por los próximos años

nera efectiva. En el resto de esta sección se presentan algunos de los esquemas criptográficos post-cuánticos.

6.2.2.1. Criptografía post-cuántica basada en códigos

Se define al peso de Hamming $w(u)$ de $u \in \mathbb{F}_{q^n}$ como el número de componentes de u que no son cero. La distancia de Hamming entre $u, v \in \mathbb{F}_{q^n}$ es $dist(u, v) = w(u - v)$. Un código linear $\mathcal{C}[n, k]$ sobre $\mathbb{F}_q$ es un vector de k dimensiones en el subespacio de $\mathbb{F}_{q^n}$. Un código puede ser definido por una matriz generadora $G \in (\mathbb{F}_q)^{k \times n}$ o por una matriz de verificación de paridad $H \in (\mathbb{F}_q)^{r \times n}$ con $r = n - k$ tal que $C = \{uG \in \mathbb{F}_{q^n} | u \in \mathbb{F}_{q^n}\}$ o $C = \{v \in \mathbb{F}_{q^n} | Hv^T \in O^r\}$, donde $HG^T = O$. Un vector s es llamado síndrome de v, si $Hv^T = s^T$.

Las matrices generadoras y de verificación de paridad no son únicas: dada una matriz no singular $S \in (\mathbb{F}_q)^{k \times k}$ (resp. $S \in (\mathbb{F}_q)^{r \times r}$), la matriz $G' = SG$ (resp. $H' = SH$) define el mismo código que G (resp. H) en otra base. Como consecuencia, no siempre es posible obtener la forma sistemática (escalonada) $G = [I_k | M], H = [-M^T | I_r]$, donde $M \in (\mathbb{F}_q)^{k \times r}$.

Dos códigos son equivalentes si las coordenadas de sus elementos difieren por a lo más una permutación. Formalmente, un código C' generado por G' es equivalente a un código C generado por G sí y solo sí $G' = SGP$ para alguna matriz de permutación $P \in (\mathbb{F}_q)^{n \times n}$, y una matriz no singular $S \in (\mathbb{F}_q)^{k \times k}$. En esencia, se dice que $C' = CP$. La decodificación funciona de la siguiente manera:

Decodificación General Dada la tripleta de enteros positivos (n, k, t), un campo finito $\mathbb{F}_q$, un código linear $\mathcal{C}[n, k] \in (\mathbb{F}_q)^n$ definido por una matrix generadora $G \in (\mathbb{F}_q)^{k \times n}$, y un vector $c \in (\mathbb{F}_q)^n$ se pregunta si acaso existe un vector

$m \in (\mathbb{F}_q)^k$ tal que $e = c - mG$ tenga peso $w(e) \leq t$. Se sabe que encontrar tal vector e es un problema NP-completo.

Decodificación por Síndrome Dada la tripleta de enteros positivos (n, k, t), un campo finito $\mathbb{F}_q$, un código linear $\mathcal{C}[n, k] \in (\mathbb{F}_q)^n$ definido por una matriz de paridad $H \in (\mathbb{F}_q)^{r \times n}$ con $r = n - k$, y un vector $s \in (\mathbb{F}_q)^r$, se pregunta si acaso existe un vector $e \in (\mathbb{F}_q)^n$ de peso $w(e) \leq t$ tal que $He^T = s^T$. Se sabe que encontrar el vector e así definido también es un problema NP-completo.

Decodificación Permutada Resolver el problema de decodificación general o el de decodificación por síndrome para un código C que sea equivalente tras realizar la permutación a un código C' que es eficientemente decodificable, consiste en encontrar la permutación y cambio de bases entre los códigos, y utilizar el código C' como pasadizo secreto (*trapdoor*) para decodificar en C. Se cree que este problema es *suficientemente difícil* para "ciertos códigos" [90].

Decodificación Recortada Resolver el problema de decodificación general o el de decodificación síndrome para un código C que sea equivalente tras permutación a algún subcódigo recortado (una proyección) de algún código C' eficientemente decodificable, consiste en encontrar la permutación y cambio de bases entre los códigos, y usar el código C' como pasadizo secreto para decodificar C. Decidir si el código es equivalente o no al código recortado es un problema NP-completo. Existen varios esquemas criptográficos basados en los problemas de códigos descritos anteriormente, por ejemplo [92]: los criptosistemas de McEliece, Niederreiter y las firmas CFS, los cuales se describen brevemente a continuación.

Sistema McEliece [100] Este criptosistema utiliza la matriz generadora para la clave pública, por lo que para el cifrado, el secreto se multiplica por esa matriz para después agregarle la información de corrección de errores (ruido gausiano). Para el descifrado, se recupera la información del vector de errores para poder eliminarlos del mensaje recibido.

Sistema Niederreiter [103] A diferencia del anterior, en este caso se utiliza la matriz de permutación como clave pública, por lo que la transpuesta del vector de corrección de errores (que contiene el mensaje) se multiplica por esta matriz. Se realiza un proceso similar en el descifrado.

Firmas CFS [94] En este esquema, la clave pública también se genera a partir de la matriz de permutación, contándose con el apoyo de un oráculo aleatorio que ayuda a encontrar un síndrome decodificable con el mensaje. De igual manera, se extrae el vector de corrección, el cual contiene el mensaje, y se utiliza como firma junto con un valor proporcionado por el oráculo aleatorio. Para la verificación, se multiplica la firma con la matriz de permutación y se corrobora con una aplicación de la firma sobre el mensaje (y el valor del oráculo aleatorio).

Retos La criptografía post-cuántica basada en códigos, propone seleccionar un tipo de código que permita que la decodificación permutada sea un problema difícil. Debido a su estructura, sufre del elevado espacio de almacenamiento y transmisión requeridos para la llave pública. La tendencia actual en investigación para este tipo de criptografía se enfoca a reducir la llave pública, y en la elaboración de protocolos criptográficos que utilicen códigos como su primitiva principal. Se considera que los esquemas basados en códigos son una de las

alternativas más promisorias para construir un criptosistema cuántico eficiente [92].

6.2.2.2. Criptografía post-cuántica basada en ecuaciones cuadráticas multivariables

En este criptosistema [93], la clave pública es una secuencia polinomial P_1, $P_2, \ldots, P_{2b} \in F_2[w_1, \ldots, w_{4b}]$, donde b es el número de bits de seguridad deseada, y las $4b$ variables $w_1, \ldots, w_{4b}$ tienen coeficientes en $\mathbb{F}_2$. Cada polinomio P_i es de a lo más grado 2, sin términos cuadrados, y es representado como una secuencia de $1+4b+\frac{4b(4b-1)}{2}$ bits. La clave pública es grande ya que tiene $16b^3+4b^2 2+2b$ bits, sin embargo, la firma de un mensaje es corta ($6b$ bits), pues únicamente consiste en los $4b$ valores de $w_1, \ldots, w_{4b} \in \mathbb{F}_2$, y una cadena r de $2b$ bits que satisfagan la identidad:

$$H(r, m) = (P_1(w_1, \ldots, w_{4b}), \ldots, P_{2b}(w_1, \ldots, w_{4b}))$$

donde H es una función hash estándar. El proceso de verificación de la firma es sencillo, ya que únicamente se necesitan b^3 operaciones de bit para evaluar los polinomios $P_1, \ldots, P_{2b}$. La seguridad del criptosistema post-cuántico multivariable descansa en la dificultad de encontrar una secuencia $w_1, \ldots, w_{4b}$ que produzca los polinomios P_1 a P_{2b} mencionados anteriormente. La evaluación de cualquier secuencia candidata tiene un costo computacional bajo, sin embargo, la probabilidad de encontrar la secuencia correcta por búsqueda aleatoria es menor a 2^{-2b}. La principal ventaja de este criptosistema es que la firma es mucho más pequeña que la generada por otros esquemas criptográficos post-cuánticos.

Por su parte, la entidad signataria logra firmar eficientemente mediante el uso de una estructura predefinida para la generación de los $2b$ polinomios P_i. A

esta estructura se le conoce como el problema de las ecuaciones de campo escondidas (HFE, *Hidden Field Equations* por sus siglas en inglés), lo que le permite al firmante resolver la ecuación en un tiempo razonable. Se cree que es posible que un atacante pudiera detectar la estructura subyacente en el polinomio, sin embargo, no se conoce hasta el momento ningún algoritmo que logre resolver HFE eficientemente, por lo que se sigue considerando como un problema difícil.

Para la selección de la estructura, se fija un polinomio irreducible $\varphi \in \mathbb{F}_2[t]$ de grado $3b$. Se define L como una extensión de campo $\mathbb{F}_2[t]/\varphi$ de tamaño 2^{3b}. La parte más interesante en el proceso de firma, es encontrar las raíces de un polinomio secreto de una sola variable de grado pequeño sobre el campo L, específicamente, un polinomio en $L[x]$ de grado a lo más $2b$.

Este polinomio secreto se escoge aleatoriamente con la forma $2^i + 2^j$, con $j \geq 0$. Si un elemento $x \in L$ se expresa con la forma $x_0 + x_1 t + \ldots + x_{3b-1} t^{3b-1}$, con $x_i \in \mathbb{F}_2$, entonces $x_2 = x_0 + x_1 t^2 + \ldots + x_{3b-1} t^{6b-2}$, $x^4 = x_0 + x_1 t^4 + \ldots + x_{3b-1} t^{12b-4}$, etc, de tal manera que $x^{2^i+2^j}$ sea un polinomio cuadrático sobre las variables $x_0, \ldots, x_{3b-1}$, después se aplican algunas transformaciones simples para ocultar la estructura. La clave pública tendría entonces tres componentes, a saber: Una matriz invertible S de $4b \times 4b$ con coeficientes en $\mathbb{F}_2$. Un polinomio $Q \in L[x, v_1, v_2, \ldots, v_b]$, donde cada elemento tiene una de las siguientes formas: $lx^{2^i+2^j}$ con $l \in L$, $2^i < 2^j$, $2^i + 2^j \leq 2b$; $lx^{2^i} v_j$, con $l \in L$ y $2^i \leq 2b$; $lv_i v_j$;lx^{2^i} ylv_j. Y una matriz T de $2b \times 3b$ de rango $2b$ con coeficientes en $\mathbb{F}_2$.

Estos tres componentes están organizados de la siguiente forma: Se forma un vector con las variables x_i y v_i como: $(x_0, x_1, \ldots, x_{3b-1}, v_1, v_2, \ldots, v_b) \leftarrow S \times (w_1, \ldots, w_{4b})$. Dentro del anillo de cocientes $L[w_1, \ldots, w_{4b}]/(w1^2 - w1, \ldots, w_{4b}^2 - w_{4b})$, se calcula $x = \sum x_i t^i$, $y = Q(x, v_1, v_2, \ldots, v_b)$, reexpresándose la variable y

como $y_0 + y_1 t + \cdots + y_{3b1} t^{3b-1}$ con cada $y_i \in \mathbb{F}_2[w_1, \ldots, w_{4b}]$. Finalmente se computa $(P_1, P2, \ldots, P_{2b}) \leftarrow T \times (y_0, \ldots, y_{3b-1})$.

6.2.2.3. Criptografía post-cuántica basada en retículas

Una retícula es un conjunto de puntos en un espacio n-dimensional con una estructura periódica. Dados n vectores b_i linearmente independientes en $\mathbb{R}^n$, se produce una retícula denotada como:

$$L(b_0, \ldots, b_{n-1}) = \left\{ \sum_{i=0}^{n-1} x_i b_i : x_i \in \mathbb{Z} \right\}$$

donde los vectores $b_0, \ldots b_{n-1}$ son la base vectorial de la retícula[99].

La criptografía post-cuántica basada en retículas basa su seguridad en la presunción de la dificultad que supone resolver diversos problemas definidos sobre éstas. Entre los principales problemas difíciles se encuentran el problema del vector más Corto, el Problema del vector más cercano, y el problema de los vectores independientes más cortos. Se considera que los tres problemas tienen el mismo grado de complejidad computacional [102]. En particular, hallar la solución al problema del vector corto, toma una complejidad temporal de $2^{O(n)}$, pero también espacio exponencial, volviéndolo impráctico. Existen también algoritmos que resuelven este problema con complejidad en espacio polinomial, pero su complejidad temporal se vuelve $2^{O(n \log n)}$. Hasta el día de hoy, sólo se conocen ataques exponenciales o con resultados de aproximación muy malos.

La criptografía basada en retículas tiene dos vertientes: los esquemas eficientes que necesitan mejores pruebas de seguridad, o los esquemas con alta probabilidad de seguridad. De estos últimos, pocos son lo suficientemente eficientes para ser competitivos con los esquemas criptográficos clásicos.

Aunque existen propuestas de algoritmos cuánticos para resolver los problemas de las retículas, estos se desempeñan peor que los algoritmos de computación clásica, por lo que es muy probable que la criptografía basada en retículas sea resistente a ataques formulados desde computadoras cuánticas.

Esquemas criptográficos El sistema criptográfico GGH es una analogía al sistema McEliece descrito anteriormente, pero basado en retículas [95]. El sistema funciona de la siguiente manera: con la finalidad de proveer una mejor eficiencia, la llave privada B es una "buena" base vectorial de retícula , conteniendo vectores cortos y casi ortogonales. En cambio, la llave pública H es una "mala" base vectorial para la misma retícula. En otras palabras, H es la peor base posible para la retícula en la que esté trabajando el criptosistema.

El cifrado consiste en introducir ruido gausiano a un punto v previamente seleccionado de la retícula a través de un pequeño vector r. En particular, se prefiere que el vector $r+v$ tenga coordenadas que estén reducidas y que descanse a lo largo de la diagonal de H (en su forma Hermitiana), de tal manera que el componente original pueda ser escogido arbitrariamente.

El proceso de descifrado consiste en encontrar el punto v de la retícula que sea más cercano al texto cifrado $c = (r \bmod H) = v + r$, y al vector de error asociado $r = c - v$. Con ello se busca que el vector de error (ruido) sea lo suficientemente corto para que el punto v pueda ser recuperado con la base privada B al utilizar el algoritmo de redondeo ("*Rounding Off*") de Babai: $v = B\lfloor B^{-1}(v+r)\rceil$.

Otro esquema criptográfico basado en retículas y propuesto para criptografía de llave pública es NTRU [97]. Este sistema está basado en anillos, los cuales pueden ser descritos equivalentemente mediante cierto tipos de retículas. Supóngase que T es una matriz de permutación que rota cíclicamente las coordenadas de un punto en una retícula, entonces $T * v$ es la matriz circulante

del vector $v \in \mathbb{Z}^n$. En el esquema NTRU, se utilizan las retículas modulares convulocionales, las cuales presentan las siguientes características: Son cerradas bajo una transformación lineal que toma el vector (x, y), conteniendo vectores n-dimensionales, y lo lleva a (Tx, Ty). En esencia, es el vector resultante de aplicar paralelamente la rotación cíclica con la matriz de permutación T antes mencionada. Son retículas q-arias (contienen $q\mathbb{Z}^{2n}$ como subretícula), por lo que la membresía de (x, y) depende exclusivamente de que (x, y) mód q estén en la retícula. Para fines prácticos, q es una potencia de 2, adicionalmente, existe un parámetro $p = 3$. El sistema funciona como sigue:

La llave privada es un vector $(f, g) \in \mathbb{Z}^{2n}$. La retícula asociada a la llave privada y q, el parámetro del sistema, se definen como: $\Lambda_q((T * f, T * g)^T)$, esto es, la retícula modular convulatoria más pequeña que contiene a nuestro vector. Los vectores secretos f, g tienen las siguientes restricciones:

- obviamente la matrix $[T * f]$ debe ser invertible módulo q
- $f \in e_1 + p, 0, -p^n$ y $g \in e_1 + p, 0, -p^n$ deben ser polinomios aleatorios tales que $f - e_1$ y g tengan $d_f + 1$ coeficientes positivos y d_f negativos (y el resto en cero).

Note que $[T * f] \equiv I(\text{ mód } p)$ y $[T * g] \equiv O(\text{ mód } p)$, donde O es la matriz nula con todas sus entradas en cero.

La llave pública, similarmente al criptosistema discutido anteriormente, corresponde a la base en su forma normal Hermitiana de la retícula modular convolucionada $\Lambda_q((T * f, T * g)^T)$ definida como parte de la llave privada. Dada la estructura de este tipo de retículas, y la selección de f, la base pública en su

forma normal Hermitiana tiene una forma particularmente sencilla:

$$H = \begin{bmatrix} I & O \\ T * h & q \cdot I \end{bmatrix}$$

donde $h = [T * f]^{-1} g (\text{mód} p)$, por lo que sólo es necesario publicar $h \in \mathbb{Z}_q^n$.

Para el cifrado, un mensaje se codifica como un vector $m \in [1, 0, -1]^n$, con exactamente $d_f + 1$ elementos positivos y $d - f$ elementos negativos. A este vector se le concatena un segundo vector $r \in [1, 0, -1]^n$ con las mismas características, dando como resultado un vector $(-r, m) \in [1, 0, -1]^{2n}$, después se reduce el vector por la matrix H (la llave pública), resultando en un vector:

$$\begin{bmatrix} O \\ (m + [T * h]r) \bmod q \end{bmatrix}$$

dado que la primer parte del vector es ceros, el mensaje se envia comprimido.

El descifrado consiste en multiplicar el mensaje cifrado por la matriz $[T * f]$

6.3. Criptografía basada en isogenias

Una isogenia es mapa racional no constante o un morfismo entre dos variedades Abelianas, por ejemplo: curvas elípticas, es surjetivo y tiene un núcleo finito. El grado de una isogenia es su grado como mapa racional. Para isogenias separables, tener un grado l, significa tener un núcleo de grado l. Cada isogenia de orden mayor a 1 puede ser factorizada en una composición de isogenias de grado primo.

Un endomorfismo de una curva elíptica E definida sobre $\mathbb{F}_q$ es una isogenia $E \to E$ definida sobre $\mathbb{F}_{q^m}$ para un valor de $m \in \mathbb{N}$. El conjunto de endomorfismos de E junto con el mapa en cero forman un anillo bajo las operaciones de

adición de punto a punto y composición; a este anillo se le conoce como anillo endomórfico de E, y se denota como End(E). Este anillo es isomórfico al orden en un álgebra quaterniana o a un orden en un campo cuadrático imaginario; en el primer caso, se dice que la curva E es supersingular, en el segundo, ordinaria (o no-supersingular).

Dos curvas elípticas definidas sobre $\mathbb{F}_q$ son isogeneas sobre $\mathbb{F}_q$ si existe una isogenia $\phi : E_1 \to E_2$ definidas sobre el mismo campo. Dos curvas son isógenas sobre $\mathbb{F}_q$ si tienen el mismo orden. Dado que cada isogenia tiene una isogenia dual, la propiedad de isogeneidad sobre $\mathbb{F}_q$ es una relación de equivalencia en el conjunto finito de clases isomórficas sobre $\bar{\mathbb{F}}_q$ de las curvas elípticas definidas sobre $\mathbb{F}_q$. Por lo que, definimos una clase isogenea como una clase de equivalencia entre curvas elípticas tomadas hasta el isomorfismo sobre $\bar{\mathbb{F}}_q$, bajo esta relación de equivalencia.

Las curvas sobre la misma clase isogenea son todas supersingulares o todas ordinarias. Hoy en día se utilizan las curvas no supersingulares para fines criptográficos, sin embargo, en este tipo de criptografía se utilizan las otras.

Todas las curvas supersingulares están definidas sobre $\mathbb{F}_{p^2}$, y por cada primo $l \nmid p$, existen $l + 1$ isogenias de grado l originandose de diversas curvas supersingulares. Dada una curva elíptica E y un subgrupo finito Φ de E, hay, hasta el isomorfismo una isogenia única $E \to E'$ con núcleo igual a Φ. En otras palabras, podemos identificar una isogenia dado su kernel, y analógicamente, dado el núcleo de un subgrupo, la isogenia correspondiente puede ser calculada.

A diferencia de las curvas ordinarias, en las curvas supersingulares hay un numero finito de clases isomórficas de curvas en cualquier clase isogenea; de hecho, este número es el genus g de la curva modular $X_0(p)$ (donde p es un número primo que define el campo base que soporta curva), el cual es $\frac{p+1}{12}$.

6.3.1. Esquema criptográfico

Sea $\mathbb{F}_q = \mathbb{F}_{p^2}$, p es un primo de la forma $l_A^{e_A} l_B^{e_B} \cdot f \pm 1$, donde $l_{A,B}$ son primos pequeños, y f es un cofactor para hacer p un primo. Para cada $l_A^{e_A}$ y $l_B^{e_B}$ se prueba una f aleatoria hasta encontrar un primo p del tamaño deseado. Dado p es fácil encontrar una curva $E/\mathbb{F}_{p^2}$ con cardinalidad $(p \mp 1)^2 = (l_A^{e_A} l_B^{e_B} \cdot f)^2$. Una vez encontrada E, se puede escoger una curva aleatoria E_0 recorriendo el gráfico de isogenias soportadas por la misma (o escoger directamente a E), la cual tendrá como estructura: $(\mathbb{Z}/(p \mp 1)\mathbb{Z})^2$. Para encontrar una base $E_0[l_A^{e_A}]$, se escoge un punto $P \in_R E_0$, y se multiplica por su cofactor: $(l_B^{e_B} f)^2$, además se escoge un punto Q con las mismas características pero que deberá ser linearmente independiente en $E_0[l_A^{e_A}]$. Uno puede verificar la linearidad de los punto aplicando el emparejamiento bilinear de Weil sobre curvas supersingulares.

Además, se escoge $\mathcal{H} = H_k : k \in K$ como una función hash indexada por un campo finito K, donde cada H_k es una función de $\mathbb{F}_{p^2}$ a un espacio de mensajes $\{0,1\}^w$.

Generación de claves Se escogen los elementos m_A,$n_A \in \mathbb{Z}/l_A^{e_A}\mathbb{Z}$ procurando que alguno no sea divisibles por l_A. Se selecciona la curva E_A y se aplican las isogenias $\phi_A(P_B)$ y $\phi_A(Q_B)$, y se escoge el elemento $k \in K$. La llave pública es $(E_A, \phi_A(P_B), \phi_A(Q_B), k)$, mientras que la clave privada es (m_A, n_A, k).

Cifrado Dada la clave pública, y un mensaje $\{0,1\}^w$, se escogen dos elementos aleatorios m_B,$n_B \in \mathbb{Z}/l_B^{e_B}\mathbb{Z}$, procurando que alguno no sea divisible por l_B, y se calcula:

$$h = H_k(j(E_{AB})),$$

$$c = h \oplus m.$$

donde $j(.)$ es el j-invariante de la curva. El texto cifrado es: $(E_B, \phi_B(P_A), \phi_B(Q_A), c)$

Decifrado Dado el texto cifrado, y la clave privada, se calcula el j-invariante de la curva E_{AB} y después:

$$h = H_k(j(E_{AB})),$$

$$m = h \oplus c.$$

donde m es el texto cifrado.

Capítulo 7

Protocolos de Seguridad

Resumen

Un protocolo de seguridad fija las pautas de actuación para unificar criterios en materia de ciberseguridad y debe estar consensuado entre los distintos actores que intervienen para ejecutarlo.

Este capítulo despliega los enfoques existentes para el diseño de protocolos de seguridad, enfatizando el enfoque formal y el enfoque de la complejidad computacional. Posteriormente se listan ejemplos de este tipo de reglas o pautas resaltando los ataques que mitigan.

7.1. Introducción

La seguridad computacional es un importante requerimiento en la vida diaria. Los crímenes como robos de identidades, accesos no autorizados a información ya han causado importantes pérdidas, en especial cuando se trata de transacciones electrónicas. Un protocolo de seguridad generalmente es usado para lidiar con estos problemas.

Un *protocolo de seguridad* es un conjunto de reglas y convenciones donde uno o mas agentes se identifican unos a otros, para poder compartir información privilegiada [126]. La meta de los protocolos de seguridad es intercambiar información privilegiada a través de redes inseguras como es la Internet. La información debe ser protegida de *hackers* en la red usando algoritmos criptográficos como cifrado simétrico y asimétrico, funciones hash y firmas digitales, todas revisadas en los capítulos anteriores. En algunos casos las marcas de tiempo de los dispositivos también permiten cumplir las metas de la seguridad informática como autentificacíon, confidencialidad, integridad y no repudio.

Aunque los algoritmos criptográficos existan, estos no garantizan la seguridad en un sistema de comunicación. Adicionalmente si un protocolo de seguridad no es diseñado cuidadosamente, podría contener fallas, las cuales pueden ser puntos de inicio para posibles ataques. Estas fallas pueden ser muy sutiles y difíciles de encontrar. Existe un número de protocolos en la literatura cuyas fallas no fueron encontradas por un largo tiempo y que han recibido un análisis intensivo. Uno de estos ejemplos es el bien conocido protocolo de llave pública de Needham y Schroeder publicado en 1978. A pesar de que es un protocolo muy simple, se encontró un defecto 17 años después de su publicación.

Para que el lector tenga un panorama general de los diferentes enfoques para diseñar y construir protocolos de seguridad la siguiente sección introduce estos

temas. Posteriormente, para entender con mayor detalle el concepto de Protocolo de seguridad y su especificación se describe una notación ampliamente utilizada en la literatura; describiendo algunos ejemplos.

7.2. Enfoques para el diseño de protocolos de seguridad

El proceso de desarrollo de *software* es una metodología impuesta para el desarrollo de una aplicación de *software* con calidad. Hay varias metodologías de desarrollo de éste, cada uno describe enfoques para una variedad de tareas o actividades que tienen lugar durante el proceso. El más conocido y más antiguo es el modelo de cascada, que consiste en los siguientes niveles: análisis de requerimientos, especificación, diseño, construcción (aplicación o codificación), integración, pruebas y mantenimiento.

En particular, cuando el software es crítico para la seguridad, el uso de modelos matemáticos es altamente recomendable. Estos modelos resuelven problemas de *software* (y *hardware*) principalmente en la especificación, diseño y prueba de los niveles. Debido a que los protocolos de seguridad son aplicaciones de seguridad críticas, los investigadores han centrado sus esfuerzos en la verificación de protocolos de seguridad con el fin de reducir los riesgos después de la implementación. Por lo que, dos enfoques distintos han surgido [127]: una de ellas se fundamenta en métodos formales y la otra en complejidad y probabilidad computacional.

7.2.1. Enfoque de métodos formales

En el enfoque de los métodos formales, tanto el programa como su especificación se modelan utilizando algunos lenguajes formales, no necesariamente el mismo. El razonamiento matemático riguroso se utiliza para demostrar que el programa y sus especificaciones están relacionados, por ejemplo, por algún tipo de equivalencia o implicación lógica. Con respecto a los protocolos de seguridad, los métodos formales se utilizan para modelar la función de cada protocolo participante, junto con las propiedades de seguridad para ser verificado. Entonces, la determinación de si un protocolo es defectuoso o no equivale a demostrar que el comportamiento previsto del protocolo puede ser logrado independientemente de las capacidades de un intruso. El modelo del intruso por lo general es Dolev-Yaho [137] pero también se consideran algunas extensiones en sus capacidades.

Fundamentalmente, los métodos formales adoptan el supuesto de la criptografía perfecta; bajo esta consideración, no hay manera de recuperar M o K solo del hecho de saber $\{|M|\}_K$. Un protocolo se considera libre de ataques si las propiedades de seguridad siendo verificadas se mantienen.

7.2.2. Enfoque de la complejidad computacional

En este enfoque las operaciones criptográficas se consideran funciones en cadenas de bits. En términos generales y siguiendo lo dicho por [127], un esquema de cifrado es definido por una tercia de algoritmos $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$. El algoritmo $\mathcal{K}$ es un generador de llaves que crea decisiones aleatorias, generando la cadena k (la llave). $\mathcal{E}$ es un algoritmo de cifrado que relaciona las cadenas k y m (el texto plano) en una cadena $\mathcal{E}_k(m)$ (el texto cifrado). El algoritmo $\mathcal{D}_k(c)$ es un algoritmo de decifrado que relaciona las cadenas k y c (el texto cifrado) en

una cadena que recupera m. Se espera que $\mathcal{D}_k(\mathcal{E}_k(m)) = m$ sea la adecuada para k y m.

Un adversario se modela esencialmente como una máquina de Turing que tiene acceso a un oráculo. El oráculo tiene como objetivo encontrar las k y r adecuadas que son capaces de descifrar un determinado texto cifrado $\mathcal{E}_k(m)$. Para realizar esta tarea, el oráculo por lo general tiene algunas pistas para facilitar su trabajo tales como el conocimiento de algunos componentes de m, el conocimiento de otros mensajes cifrados bajo la misma llave k, etc. Un protocolo se considera bueno si el oráculo no puede encontrar k, o mientras se consume la potencia de cálculo a mano, la probabilidad de encontrar k es de bajo crecimiento bajo un umbral determinado.

7.2.3. Conexion entre ambos puntos de vista

Como se menciona en [128], las conexiones entre el punto de vista formal y el computacional han sido focos de investigación. Estos trabajos han empezado independientemente con [129] y [127]. En particular, el papel fundamental de Abadi and Rogaway [127] fue establecer una relación de equivalencia sintáctica entre los significados de *mensaje*, donde el cifrado es tratado como un operador formal y seguro. El resultado de Abadi y Rogaway es aplicable para el cifrado simétrico en la presencia de un intruso pasivo. Algunas conexiones semánticas entre estos puntos de vista han sido desarrolladas por el equipo de IBM en Zurich, Ej. [130]. Herzog, [132, 131], muestra que si un ataque a un protocolo existe en el modelo formal, existe un ataque en el modelo de complejidad computacional. Posteriormente Baudet et. al. [133] introdujo un framework racional para proveer implementaciones de teorías ecuacionales, las cuales son usadas para especificar primitivas criptográficas. Recientemente, Kremer y Mazare [134] han

extendido el trabajo de Baudet et al. al considerar un usuario adaptativo en vez de uno completamente pasivo.

Este capítulo se perfila en el enfoque de los métodos formales, así que en la siguiente sección, se describirá en un nivel mas detallado el enfoque formal para el protocolo de verificación de seguridad.

7.2.4. Corrección Automatizada de Protocolos de Seguridad

Como se ha mencionado en secciones previas han habido varios enfoques por identificar vulnerabilidades en protocolos de seguridad. Adicionalmente a estos trabajos surge otra línea de investigación enfocada a corregir protocolos de seguridad de manera automática [135].

Un ejemplo de esta línea es SHRIMP por sus siglas en inglés *Smart metHod for Repairing IMperfect security Protocols* tiene como objetivo acelerar el ciclo de desarrollo de protocolos de seguridad, uniendo la brecha que hay entre diseño y análisis por medio de una tarea de diagnóstico y reparación automatizada. SHRIMP trabaja con herramientas de verificación existentes para el proceso de búsqueda de ataques en el protocolo. Primero analiza un protocolo y, si el protocolo tiene fallas, busca una o más corridas del protocolo que violan un requerimiento de seguridad, llamado *ataque*. Así, SHRIMP analiza el protocolo y el ataque para indicar qué pasos fallan en el protocolo y de esta forma sintetizar cambios apropiados para corregirlos; produciendo una versión mejorada del protocolo, el cuál es analizado y parchado nuevamente hasta que no sean detectados más errores.

SHRIMP es un conjunto de métodos de parche (basado en la metodología *proof planning*), cada uno de ellos es capaz de parchar una clase general de fallas.

Para arreglar una falla, cada método de parche se basa en los principios de Abadi y Needham. Los casos de éxito mostrados son los referentes a parchar protocolos vulnerables a los ataques de tipo *replay*. La herramienta fue probada con 36 protocolos obteniendo una tasa de reparación del 90 %.

7.3. Métodos formales en protocolos de seguridad

En general, los métodos formales pueden ser aplicados en varios puntos durante el proceso de desarrollo. Desde una perspectiva de alto nivel, la comunidad de los métodos formales para los protocolos de seguridad ha concentrado sus esfuerzos en la especificación y verificación de procesos. De forma superficial, la tarea de verificar los protocolos de seguridad actualmente requiere de dos pasos principales:

Especificación formal: En primer lugar un protocolo debe traducirse a partir de una especificación informal a una especificación formal, esto usando cierta lógica. En este paso las propiedades que deben verificarse también son formalizadas. Este es un paso clave y requiere la intervención humana.

Verificación: Una vez formalizado, el protocolo y la propiedad a ser verificada se envían a un analizador. Si es basado en búsqueda, el analizador puede producir una de dos salidas : OK , lo que indica que el protocolo está libre de errores, o de un script, señalando que el protocolo es defectuoso. Este script es un contraejemplo: una intercalación de uno o más protocolos se ejecuta, violando la propiedad de seguridad. Como veremos en la siguiente sección, la mayoría enfoques de verificación incluyen un buscador de contraejemplos automático.

Entonces, determinar si un protocolo es defectuoso o no depende en las propiedades de seguridad que estén siendo verificadas.

7.3.1. Propiedades de seguridad

En general, la comunidad de métodos formales se ha enfocado en dos propiedades de seguridad: Autentificación y Privacidad.

- **Autenticación**: un agente debe estar seguro de la identidad de otros agentes a través de un intercambio de mensajes. Esta propiedad tiene distintos significados, aunque, Lowe [136] ha propuesto una jerarquía de cuatro niveles de seguridad para interpretar formalmente esta propiedad.
- **Privacidad**: Ciertas partes de los mensajes (llamados también secretos o confidenciales) en la red solo pueden ser leídos por sus destinatarios.

El enfoque de métodos formales diseña un modelo de intruso con ciertas capacidades. Si el protocolo es defectuoso, el intruso puede romper una o más de las propiedades de seguridad siendo verificadas.

7.3.2. Habilidades del intruso

Dolev y Yao, [137], han formalizado un modelo de espía el cual se ha convertido en un estándar de referencia en la literatura. En particular destacan las siguientes habilidades descritas:

- No puede hacer criptoanálisis.
- Es capaz de ver todo el tráfico en la red.
- Puede retrasar mensajes.

- Puede prevenir que los mensajes lleguen a su destinatario.
- Puede dividir mensajes que ha visto en el tráfico.
- Puede agregar falsos mensajes.
- Puede enviar los mensajes que no puede leer (texto cifrado).
- Conoce todas las llaves públicas, sus propias llaves pública y privada y las llaves privadas de agentes que han sido comprometidos.

7.4. Notación para la especificación de protocolos

El objetivo de esta sección es presentar a nuestro lector un conjunto de símbolos y convenciones que usaremos para especificar protocolos de seguridad.

7.4.1. Descripción de protocolos

La representación de un protocolo consiste en dos partes: (a) conocimiento inicial de los agentes participantes, y (b) los eventos que componen el protocolo, que denominaremos transiciones (ver Tabla 7.1).

7.4.1.1. Conocimiento inicial

Todo agente inicialmente conoce, al menos, su identidad. El conocimiento inicial de un agente debe especificarse como sigue:

$$\mathsf{Ag}_1 : [ik, ik2, ...]$$

Abreviación	**Descripción**
Conocimiento Inicial:	
$\mathsf{Ag}_1 : [ik]$	Agente Ag_1 conoce ik.
$\mathsf{Ag}_2 : [ik, ik2]$	Agente Ag_2 conoce ik e $ik2$.
Transiciones:	
$\mathsf{Ag}_1 \rightarrow \mathsf{Ag}_2 : m_1$	Ag_1 envía mensaje m_1 a Ag_2, el cual lo recibe y de él obtiene
$m2 = f(m_1)$	$m2$ después de un **proceso local** evaluando $f(m_1)$.
$\mathsf{Ag}_2 \rightarrow \mathsf{Ag}_1 : m_2$	Ag_2 envía m_2 a Ag_1, el cual lo recibe.

Tabla 7.1: Descripción de un Protocolo

En caso contrario, consideraremos que:

$$\mathsf{Ag}_1 : [\quad] \qquad \simeq \qquad \mathsf{Ag}_1 : [\mathsf{Ag}_1]$$

7.4.1.2. Transiciones

Sean Ag un símbolo que denota un agente, $\rightarrow$ otro que denota un canal de comunicaciones, y m una representación de un mensaje. Para representar una transición usamos esta notación:

$$\mathsf{Ag}_1 \rightarrow \mathsf{Ag}_2 : m$$

Esto representa el evento de comunicación del mensaje m del emisor Ag_1 al receptor Ag_2. Note que los procesamientos locales tienen lugar justo después de que un agente recibe un mensaje, y es denotado como la evaluación de una función, $(f(m))$.

7.4.2. Mensajes

Un mensaje m puede ser *atómico* o *compuesto*. Un mensaje atómico es aquel que no puede descomponerse en más mensajes, mientras que un mensaje compuesto contiene otros mensajes, que pueden ser compuestos o atómicos. En la Tabla 7.2 resumimos la notación que usamos para representar mensajes.

Tipo	**Abreviación**	**Descripción**
	m	Un mensaje cualquiera.
	;	concatenación de mensajes.
Atómico	s	Un mensaje en texto plano, s es una cadena.
	$F(m)$	Función de sólo un sentido, por ejemplo Hash(m) or Timestamp(m)
	$Key(s)$	Una llave *Key*, con identificador s
Compound	$\{\!\lvert m \rvert\!\}_{Key(s)}$	Un mensaje, resultado de cifrar m con la llave $Key(s)$
	$m_1; m_2$	Un mensaje, resultado de concatenar los mensajes m_1 y m_2

Tabla 7.2: Tipos de mensajes

La siguiente notación es usada para representar cifrado simétrico y cifrado público.

- Cifrado simétrico. Una misma llave es usada en el proceso de cifrado como de descifrado de un mensaje:

$$\begin{aligned} X &= \{\!\lvert m \rvert\!\}_K \\ m &= K^{-1}(X) \end{aligned} \tag{7.1}$$

- Criptografía asimétrica. Cada agente (e.g. Ag_1) tiene un par de llaves (una pública, $K_{pub(\mathsf{Ag}_1)}$, y otra privada, $K_{priv(\mathsf{Ag}_1)}$. Por convención, consideramos que la llave privada sólo es del conocimiento del agente en cuestión.

Usualmente, dicho agente cifra mensajes usando su llave privada, los cuales podrán ser leídos únicamente por aquellos otros agentes que conozcan la llave pública de dicho agente. Para garantizar comunicación confidencial de un agente a otro, el primero cifrará el mensaje con la llave pública del segundo; en contraste, para garantizar autenticidad de participación en la confección de un mensaje, un agente le cifrará usando su llave privada.

La Tabla 7.3 muestra la notación y el significado de diferentes tipos de mensajes atómicos, usados comúnmente en el diseño de un protocolo de seguridad.

Notación	**Descripción**
N_a, N_b, N_c	Conocidos como *nonces*, son números aleatorios, que se supone no han sido usados antes y no se pueden adivinar.
T_a, T_b, T_c	Estampas de tiempo, que especifican la hora actual (incluyendo milisegundos.)
K	Una llave
A,B,C	Agentes
$K_{[\mathsf{Ag}_1,\mathsf{Ag}_2]}$	La llave simétrica que comparten los agentes Ag_1 y Ag_2.
$K_{pub(\mathsf{Ag}_1)}$	La llave pública del agente Ag_1
$K_{priv(\mathsf{Ag}_1)}$	La llave privada del agente Ag_1
Spy	Intruso con fortalezas Dolev y Yao (Sección 7.3.2)

Tabla 7.3: Símbolos para mensajes de acuerdo a su especialización

7.5. Ejemplos de Protocolos de Seguridad

En esta sección abordaremos la descripción de dos Protocolos de Seguridad y sus ataques. Estos ataques son ejemplos de que el diseño correcto de protocolos de seguridad no es una tarea fácil.

7.5.1. Protocolo NSPK

Roger Needham y Michael Schroeder publicaron un protocolo de llave pública en 1978. El protocolo llamado NSPK por las siglas de sus autores constan de siete pasos, pero solo tres se refieren a la autenticación de los participantes.

El objetivo del protocolo NSPK es autenticar dos agentes utilizando criptografía asimétrica. El conocimiento inicial es:

$$\begin{aligned} A &: A, B, K_{pub(A)}, K_{pub(B)} \\ B &: A, B, K_{pub(A)}, K_{pub(B)} \\ Spy &: A, B, K_{pub(A)}, K_{pub(B)} \end{aligned}$$

Los pasos del protocolo son:

$$\begin{aligned} 1 \quad A &\rightarrow B : \{\!|N_a; A|\!\}_{K_{pub(B)}} \\ 2 \quad B &\rightarrow A : \{\!|N_a; N_b|\!\}_{K_{pub(A)}} \\ 3 \quad A &\rightarrow B : \{\!|N_b|\!\}_{K_{pub(B)}} \end{aligned}$$

La descripción del protocolo es la siguiente:

- En el paso 1, el agente A actuando como iniciador empieza una corrida del protocolo con el agente B, quien actúa como respondedor. A cifra el mensaje $\{\!|N_a; A|\!\}$ con la llave pública del agente B, denotado como $K_{pub(B)}$. Este mensaje contiene un *nonce* N_a construido por A como una prueba

de frescura de la sesión; además de enviar también el nombre de agente A. B descifra el mensaje, ya que él es el único que contiene la llave privada respectiva para descifrar el mensaje y allí se da cuenta que A desea iniciar una sesión con él.

- Siguiendo la especificación del protocolo, B crea un nuevo *nonce* N_b y junto con el *nonce* N_a recibido en el paso anterior crea el mensaje del paso dos cifrándolo con la llave pública de A, pués es el único que podría descifrar dicho mensaje. A descifra el mensaje enviado por B y verifica que N_a es el *nonce* que él generó al iniciar la corrida del protocolo; en ese momento A autentifica a B, pues sabe que solo B pudo haber interpretado el mensaje del paso 1. Además, también se da cuenta que recibe un nuevo desafío creado por B, en este caso N_b.

- Como todo va bien, A continúa el protocolo y responde enviando el último mensaje del protocolo. En este caso, envía a B el secreto recibido en el paso dos que es N_b; esto lo hace cifrándolo nuevamente con la llave pública de B. B al recibir este mensaje, asegura la participación de A puesto que asume que A era el único que podría haber interpretado el mensaje enviado en el paso 2. En este paso B autentifica a A y están listos ambos de utilizar N_a y N_b en futuras conversaciones.

7.5.2. Ataque al Protocolo NSPK

El protocolo NSPK parece correcto a primera vista, pero es defectuoso. Lowe, [138], descubrió que un intruso podía hacerse pasar por un agente que simultáneamente puede tener una sesión con otro agente . A continuación se puede

ver el ataque:

$$
\begin{array}{llcll}
s1:1. & A & \rightarrow & \mathsf{Spy} & : \{\!|N_a; A|\!\}_{K_{pub(Spy)}} \\
s2:1. & \mathsf{Spy}(A) & \rightarrow & B & : \{\!|N_a; A|\!\}_{K_{pub(B)}} \\
s2:2. & B & \rightarrow & \mathsf{Spy}(A) & : \{\!|N_a; N_b|\!\}_{K_{pub(A)}} \\
s1:2. & \mathsf{Spy} & \rightarrow & A & : \{\!|N_a; N_b|\!\}_{K_{pub(A)}} \\
s1:3. & A & \rightarrow & \mathsf{Spy} & : \{\!|N_b|\!\}_{K_{pub(Spy)}} \\
s2:3. & \mathsf{Spy}(A) & \rightarrow & B & : \{\!|N_b|\!\}_{K_{pub(B)}}
\end{array}
$$

En el ataque de Lowe, una instancia idéntica del mensaje 2, $\{\!|N_a; N_b|\!\}_{K_{pub(A)}}$, se usa en dos ejecuciones independientes ($s1:2$ y $s2:2$). El agente engañado, A, el iniciador de la primera ejecución, es el destinatario de la sesión 2 ($s2$), pero no puede distinguir quién construyó o envió el mensaje. Así, mientras B sabe que A ha participado recientemente en una corrida del protocolo, no puede decir si A lo está ejecutando aparentemente con él. Como se puede ver, el mensaje 2 de la descripción del protocolo no incluye el nombre del agente originador del mensaje. Así pues la mejora a este protocolo es incluir el nombre del agente originador y el respondedor, quedando el protocolo corregido de la siguiente manera:

$$
\begin{array}{llcll}
1 & A & \rightarrow & B & : \{\!|N_a; A|\!\}_{K_{pub(B)}} \\
2 & B & \rightarrow & A & : \{\!|A; B; N_a; N_b|\!\}_{K_{pub(A)}} \\
3 & A & \rightarrow & B & : \{\!|N_b|\!\}_{K_{pub(B)}}
\end{array}
$$

7.5.3. Protocolo Wide-Mouth Frog

El protocolo Wide Mouthed Frog (WMF) [139] consiste en un acuerdo de llave de sesión de criptografía simétrica, usando un servidor confiable como mediador.

El conocimiento inicial del protocolo es el siguiente:

$$\begin{aligned} A &: A, B, K_{AS} \\ B &: A, B, K_{BS} \\ \mathsf{S} &: A, B, K_{AS}, K_{BS} \\ Spy &: A, B \end{aligned}$$

A se conoce así mismo, conoce con quién quiere comunicarse B y conoce la llave que comparte con el servidor K_{AS}. Por otro lado, B también se conoce así mismo, conoce de la existencia de A y conoce la llave que comparte con el servidor K_{BS}. El servidor conoce lo que conoce A y B. En tanto que el *Spy* solo conoce los nombres de A y B. Este protocolo solo consiste de dos pasos:

$$\begin{aligned} 1 \quad A &\rightarrow \mathsf{S} : A; \{\!|B; T_a; K_{ab}|\!\}_{K_A} \\ 2 \quad \mathsf{S} &\rightarrow B : \{\!|A; T_{a+d}; K_{ab}|\!\}_{K_B} \end{aligned}$$

La explicación del protocolo es la siguiente:

- Al principio de la corrida el agente A empieza la conversación, mandando al servidor S un mensaje compuesto de dos partes: una de texto plano y otra de texto cifrado. La primera parte es utilizada para saber quien es el emisor; la segunda parte da confidencialidad sobre quien es el destinatario (en este caso el agente B), una marca de tiempo para saber cuando se creó el mensaje y la llave de sesión K_{ab} generada por A. Este mensaje es descifrado por el servidor y validad que el mensaje es reciente a través de T_a.

- En la segunda parte del protocolo una vez que el servidor valida el primer paso, manda un mensaje al destinatario (en este caso B) incluyendo en

el mensaje cifrado tres elementos: el nombre del iniciador A, una marca de tiempo actualizada T_{a+d} y la llave de sesión K_{ab} generada al inicio de la corrida. En este protocolo B confía en el servidor y en sus marcas de tiempo. Por tanto, acepta la llave de sesión recibida para posteriormente comunicarse con A.

7.5.4. Ataque al Protocolo Wide-Mouth Frog

El ataque al protocolo es el siguiente:

$$
\begin{array}{llcll}
s1:1. & A & \rightarrow & S & : A; \{\!|B; T_a; K_{ab}|\!\}_{K_A} \\
s1:2. & S & \rightarrow & Spy(B) & : \{\!|A; T_{a+d}; K_{ab}|\!\}_{K_B} \\
s2:1. & Spy(B) & \rightarrow & S & : B; \{\!|A; T_{a+d}; K_{ab}|\!\}_{K_B} \\
s2:2. & S & \rightarrow & Spy(A) & : \{\!|B; T_{a+2d}; K_{ab}|\!\}_{K_A} \\
s3:1. & Spy(A) & \rightarrow & S & : A; \{\!|B; T_{a+2d}; K_{ab}|\!\}_{K_A} \\
s3:2. & S & \rightarrow & B & : \{\!|A; T_{a+3d}; k_{ab}|\!\}_{K_B}
\end{array}
$$

En el paso 1 (sesión 1 $s1$), cuando el servidor S recibe el mensaje, él ve algo como: A, X y asume que X ha sido cifrada utilizando la llave compartida con el agente A por el agente concatenado en texto plano. Cuando el servidor decifra X, verá si el resultado es consistente (sigue la especificación del protocolo) y compone el siguiente mensaje para enviarlo a B. En ese momento el intruso lo intercepta e impide que llegue a su destinatario ($s1:2$). Notemos que la estructura del mensaje interceptado por el intruso es similar al primer mensaje, y el intruso aprovechándose de eso inicia una nueva sesión con A, pero haciéndose pasar por B, entonces envía el mensaje al servidor como si fuera una nueva sesión ($s2:1$). El servidor asume que es una nueva comunicación entre B y A ejecuta la corrida de la sesión 2 y envía a A el mensaje 2 ($S2:2$). Nuevamente el intruso atrapa el mensaje enviado por el servidor y vuelve hacer el mismo mecanismo e

inicia una nueva sesión $s3$. Al final B asume una comunicación con A, pero hay una confusión en el tipo de sesión con la que ambos A y B se están comunicando. El ataque también provoca que el intruso pueda mantener varias sesiones vivas con el servidor.

7.6. Discusión

La experiencia ha demostrado que el diseño de los protocolos de seguridad es particularmente difícil y propenso a errores. Debido a eso, la verificación de los protocolos de seguridad ha atraído un gran interés en la comunidad de métodos formales produciendo diversas herramientas y técnicas en los últimos años. Algunos de estos métodos se basan en creencias lógicas como BAN [139] y GNY [140]; otros métodos son exploración de estados, como NRL [138], FDR [141], AVISPA [142]; otros metodos usan el modelo *strand spaces model* como [143, 144]; y finalmente, otros metodos estan basados en demostración de teoremas, como el método inductivo de Paulson [126], Coral [145], etc.

Complementando el uso de métodos formales, otros autores han adoptado un enfoque diferente, proponiendo principios de diseño con el fin de ayudar en la construcción de mejores protocolos de seguridad. El raciocinio detrás de esta iniciativa proviene de las observaciones que los ataques exitosos contra los protocolos de seguridad que son el resultado de malas prácticas en el diseño de protocolo de seguridad y que los buenos principios de diseño pueden llevar a estos protocolos a ser significativamente más robustos.

Los principios de diseño más importantes para los protocolos de seguridad han sido propuestos por Abadi y Needham [146]. Abadi y Needham llegaron a estos principios al notar algunos de los errores más comunes en el diseño de pro-

tocolos de seguridad. Si se evitan estos errores, los protocolos tienden a ser más legibles y aun más importante, más acertados. Un intento por automatizar estos principios dado un ataque conocido puede ser explorado en la investigación en Shrimp, [135].

Capítulo 8

Internet de las cosas

Resumen

Hace poco tiempo las computadoras de escritorio se volvieron una herramienta esencial para la industria y las actividades diarias de cada persona. Pero el desarrollo tecnológico ha permitido que los dispositivos móviles tengan el poder de procesamiento de una computadora de escritorio y puedan conectarse entre sí por medio de *Internet*. En este último capítulo, se hace enfásis sobre la arquitectura del Internet cuando conecta muchas cosas y sobre las vulnerabilidades de seguridad que tiene este ecosistema mejor conocido como Internet de las cosas.

8.1. Introducción

La idea del Internet de las cosas, expresada inicialmente como "computadoras en todas partes", fue formulada inicialmente por Ken Sakamura en la Universidad de Tokio en 1984, y como "computación ubicua" por Mark Weiser en 1988 (Xerox PARC). Sin embargo, fue Kevin Ashton quien en 1999 utilizó por primera vez la frase "Internet de las cosas" (IoT – acrónimo del inglés Internet of Things) en el contexto de la administración de cadenas de alimentación. Posteriormente, en 2001 la idea del IoT fue desarrollada por David Brock en un artículo del Auto-ID Center del Instituto Tecnológico de Massachusetts sobre código electrónico de productos [147].

Desde entonces un gran número de investigadores ha seguido y desarrollado esta idea, plasmada en una gran variedad de artículos científicos, libros y conferencias. En todos ellos persiste la visión de integrar inteligencia en los objetos que nos rodean para dotarlos de capacidad de sensado/actuación, procesamiento e interconexión vía Internet con el fin de proveer servicios a distintos usuarios.

Es precisamente en los usuarios, donde radica la principal diferencia del IoT con el Internet tradicional, ya que mientras en el Internet tradicional los usuarios son personas conectadas al internet vía una PC, laptop o algún dispositivo móvil (teléfono inteligente, tableta, por ejemplo), en el IoT los usuarios pueden ser a su vez otros objetos o 'cosas inteligentes' que requieren de algún servicio. En el IoT se plantea la interacción incluso con fuentes adicionales de información como las redes sociales. Más aún, el concepto involucra algo que se olvida frecuentemente: el análisis, almacenamiento y la utilización de la información colectada por los distintos dispositivos. Esto abre la posibilidad de desarrollar una variedad de aplicaciones entre las que destacan las siguientes: seguimiento de mercancías y personas, casas inteligentes, ciudades inteligentes, control y

comando remoto, navegación peatonal, servicios basados en la posición, monitoreo remoto de pacientes y monitoreo del medio ambiente; por nombrar solo algunos ejemplos [148].

En la literatura existen diferentes definiciones de IoT, de todas ellas una que los autores consideran la más adecuada es la siguiente: "Interconexión de dispositivos de sensado y actuación que hacen posible la distribución de información a través de distintas plataformas por medio de una estructura unificada, desarrollando un esquema operativo común para posibilitar aplicaciones innovadoras. Esto se logra con sensado ubicuo, análisis y representación de la información, con la computación en la nube como un marco unificador."

El Internet de las Cosas (IoT, por sus siglas en inglés) representa la evolución radical del Internet actual a una red interconectada de 'objetos inteligentes' que no solamente colectan información del medio ambiente (sensado) e interactúan con el mundo físico (actuación, comando y control), sino que también utilizan el Internet para proveer servicios de transferencia, análisis, aplicaciones y comunicación de la información. Se estima que actualmente existen alrededor de 31 mil millones de dispositivos conectados a Internet. Más aún, un estudio realizado por IHS predice que para el 2025 existirán un poco más de 50 mil millones de dispositivos o "cosas" conectados al Internet [149].

La idea del IoT ha pasado a ser una realidad gracias a la convergencia de un grupo de cuatro tecnologías principalmente: Sistemas de supervisión, control y adquisición de datos (del inglés *Supervisory Control And Data Acquisition* – SCADA), radio frecuencia identificación (RFID *Radio Frequency Identification* – por sus siglas en inglés), redes de sensores inalámbricos (WSN *Wireless Sensor Network-* por sus siglas en inglés), y comunicación máquina a máquina (M2M *Machine to Machine* - por su acrónimo en inglés). Una breve descrip-

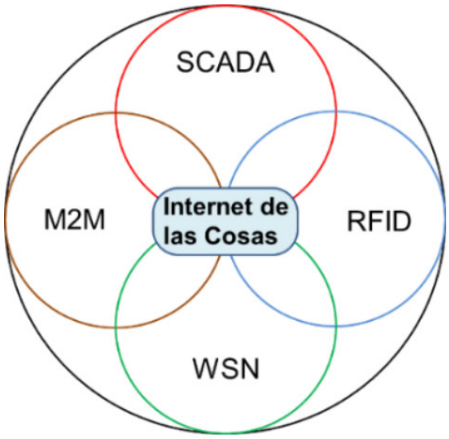


Figura 8.1: El Internet de las cosas como la convergencia de cuatro tecnologías principales

ción de cada una de estas tecnologías se da a continuación y se ilustran con la Figura 8.1

M2M. Utiliza dispositivos para capturar eventos y enviarlos a través de una red, generalmente celular inalámbrica (2G, 3G, 4G), pero existen conexiones por cable o híbridas, a otro dispositivo o aun servidor central que convierte los eventos recibidos en información de utilidad, por ejemplo, para emitir una alerta de falla.

RFID. Tecnología que utiliza ondas de radio para transferir información almacenada en una etiqueta electrónica o *tag*, fijada a un objeto, a un sistema central a través de un lector con el propósito de identificar y seguir al objeto.

WSN. Consiste en dispositivos, denominados nodos, espacialmente distribuidos y que de manera autónoma sensan el medio ambiente o algún parámetro físico, como temperatura, presión, aceleración, CO_2, y que de manera cooperativa transmiten dicha información a través de una red inalámbrica, generalmen-

te del tipo malla (*mesh*), y algunas veces por una red por cable o hibrida, a un determinado centro de colección, almacenamiento y análisis. Las WSN han demostrado su utilidad en una gran variedad de aplicaciones como monitoreo del medio ambiente, monitoreo de infraestructura, sistemas de transporte y logística, vigilancia y monitoreo de personas, localización y seguimiento de productos, monitoreo de maquinaria, monitoreo de la salud de pacientes, entre otras.

SCADA. Es un sistema autónomo basado en la teoría de control en lazo cerrado, un sistema inteligente o un sistema ciber-físico, que conecta monitores y controla equipo a través de una red, generalmente del tipo red por cable de distancia corta. Últimamente esto se hace através de una red inalámbrica o híbrida, en una instalación, tal como una planta generadora de energía eléctrica o un edificio.

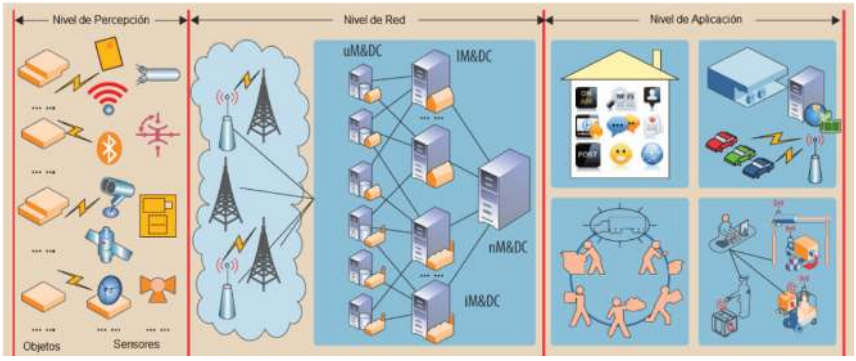
8.2. Arquitecturas de IoT

La arquitectura del IoT que es más reconocida es la que integra los siguiente tres niveles: Nivel de Percepción, Nivel de Red y Nivel de Aplicación, como se ilustra en la Figura 8.2. Una breve descripción de cada nivel se da a continuación [150].

Nivel de Percepción. En este nivel se encuentran las tecnologías que realizan el sensado, medición o detección de algún fenómeno, que se pueden fijar a objetos o dispositivos, y cuya tarea principal es la de identificar objetos y generar información. En este nivel también se incluyen actuadores mecánicos y electrónicos que ejecutan alguna instrucción para interactuar con el mundo físico.

Nivel de Red. Este nivel incluye todos los componentes de red, interfaces, enrutadores y puertas de enlace. Existen centros de gestión de datos (*Ma-*

Figura 8.2: Arquitectura del IoT



nagement and data centers - M&DCs) que actúan como nodos. Estas unidades de M&DCs están bajo la dirección directa o indirecta de centros locales (lM&DCs), industriales (iM&DCs) o nacionales (nM&DCs). En este nivel se presenta una configuración heterogénea de redes que puede incluir Internet, WSN (de área personal, local, amplia, y/o metropolitana) y redes de telecomunicaciones (2G, 3G, LTE). La función principal de este nivel es asegurar la transmisión confiable de información, así como mantener conectividad, aplicando algoritmos de codificación, fusión, búsqueda e integración de información.

Nivel de Aplicación. En este nivel se soportan aplicaciones a nivel local, industrial y nacional, gestionadas respectivamente por: lM&DCs, iM&DCs y nM&DCs. Una aplicación IoT a nivel local, interconecta IoTs en una región geográfica determinada. Una aplicación IoT industrial gestiona unidades IoT en una industria determinada, por ejemplo, sistema de transporte o telecomunicaciones. Una aplicación IoT nacional, integra elementos IoT locales e industriales de un país. En este nivel también se incluyen funciones de integración de servicios, supervisión transnacional y coordinación internacional.

8.3. Cómputo en la nube, en el borde y en la niebla

Durante años, la computación en la nube fue la tecnología base más utilizada para muchas arquitecturas empresariales generando el traslado de todos sus datos, computación, procesamiento, etc. de la infraestructura local a la propia nube. La nube parece ofrecer espacio de almacenamiento infinito y escalado para la computación sin preocupaciones desde el punto de vista de una empresa resultando en menos tiempo invertido al manejo de infraestructuras en las instalaciones y menos dinero para invertir.

La computación en la nube, como se ilustra con la Figura 8.3, permite un acceso de red conveniente y bajo demanda a un grupo compartido de recursos informáticos configurables, como redes, servidores, almacenamiento, aplicaciones y servicios, que se pueden aprovisionar y liberar rápidamente con un esfuerzo de administración mínimo o un proveedor de servicios. Sin embargo, para muchas operaciones, un modelo solo en la nube no es necesariamente el ajuste perfecto. El costo del almacenamiento en la nube está disminuyendo, pero transmitir y almacenar cantidades masivas de datos en la nube para su análisis rápidamente se vuelve prohibitivamente caro. Por lo general, todos los datos se transmiten con su fidelidad actual, sin capacidad para seleccionar qué datos son de mayor valor comercial y qué datos pueden ignorarse.

Las plataformas de IoT basadas en la nube también requieren una conexión de red constante, lo que las hace menos ideales para empresas con operaciones remotas, que no pueden permitirse interrumpir las operaciones cuando su conexión se interrumpe. Como consecuencia, se ha propuesto un nuevo conjunto de paradigmas para el manejo de datos de IoT.

Figura 8.3: Representación del cómputo en la nube (Cloud Computing)

El término informático de borde se acuñó por primera vez alrededor de 2002 y se asoció principalmente con el despliegue de aplicaciones sobre redes de entrega de contenido (CDN, *Content Deliery Network*). El objetivo principal de este enfoque era beneficiarse de la proximidad y los recursos de los servidores de borde CDN para lograr una escalabilidad masiva. Un nodo de borde incluye enrutadores, estaciones base móviles y conmutadores que enrutan el tráfico de red. En este caso, borde debe entenderse como el borde de Internet. Estos dispositivos realizan un procesamiento sofisticado para manejar los paquetes que llegan a través de las diferentes subredes que agregan. Ejemplos de dispositivos de borde son aplicaciones en la interfaz de un centro de datos que realizan funciones como la aceleración de XML, el equilibrio de carga u otro procesamiento de contenido, así como dispositivos en el punto de entrada de una empresa que realizan funciones relacionadas con la seguridad como cortafuegos, detección de intrusiones y comprobación de virus. Este concepto coloca las aplicaciones, los datos y el procesamiento en los extremos lógicos de una red en lugar de centralizarlos. Colocar datos y aplicaciones de uso intensivo de datos en el borde

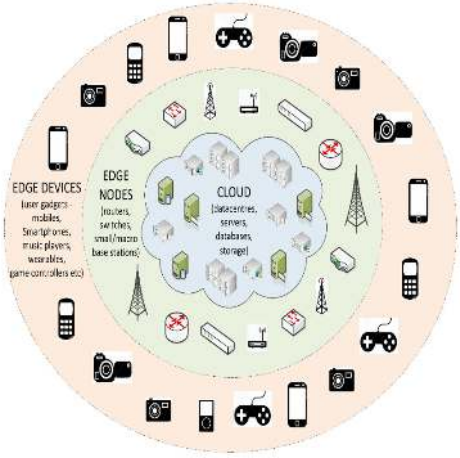


Figura 8.4: Vista simplificada del paradigma del cómputo en el borde

reduce el volumen y la distancia a la que deben moverse los dato. La Figura 8.4 ilustra esta idea.

El concepto de computación en la niebla tiene como objetivo acercar las características del servicio en la nube a lo que se conoce como "cosas", incluidos sensores, sistemas integrados, teléfonos móviles, automóviles, etc. Estas cosas forman parte de lo que en este documento se ha etiquetado como dispositivos de borde de IoT. Es importante notar que, para esta sección, cuando nos referimos a los dispositivos de borde nos referimos al borde de IoT, a menos que se indique lo contrario, tratando de mantener la esencia original de los autores de referencia pero diferenciándonos del concepto de borde de Internet.

La primera definición formal de computación en la niebla fue establecida en 2012 por Bonomi de CISCO como: "La computación en niebla es una plataforma altamente virtualizada que proporciona servicios de computación, almacenamiento y redes entre los dispositivos finales y los centros de datos tradicionales de computación en la nube, generalmente, pero no exclusivamente ubicado

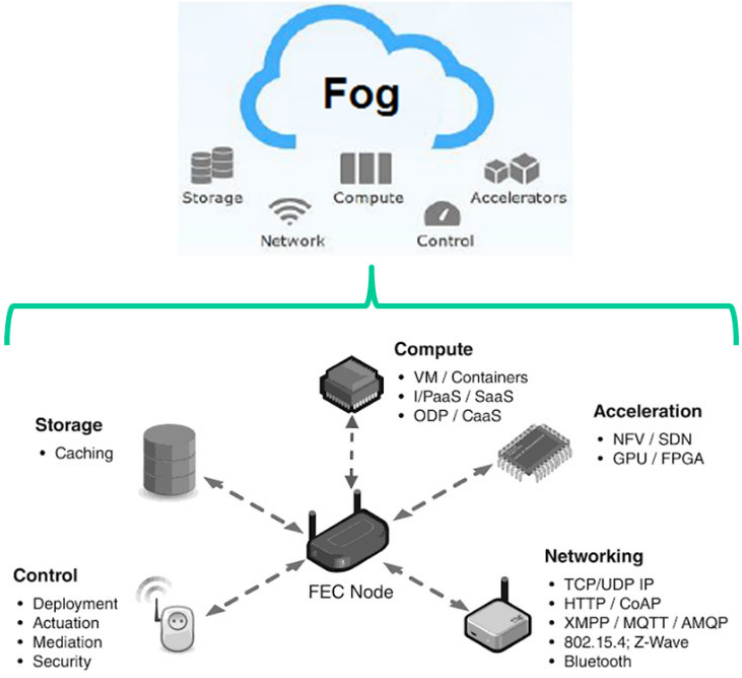


Figura 8.5: Recursos en un nodo de cómputo en la niebla (nodo fog)

en el borde de la red." Desde entonces, han surgido diferentes definiciones en distintos escenarios y contextos (Ver Figura 8.5).

8.4. Vulnerabilidades en el IoT

Es importante definir el término privacidad en una red inteligente[151]. En general, la privacidad en una red inteligente consiste en mantener la información útil de los usuarios fuera del alcance de terceros, para evitar la divulgación del perfil de vida del usuario.

En este contexto, los medidores inteligentes son un elemento clave para conocer las actividades de los usuarios al ser un instrumento de comunicación bidireccional entre el usuario y las empresas de servicios públicos [152]. Esto permite el intercambio de datos relacionados con la fijación de precios, así como la información de su consumo. Debido a su naturaleza automatizada y detallada, los datos emitidos desde un medidor inteligente pueden revelar mucho sobre las actividades y el comportamiento del usuario cuando se exponen a técnicas de minería de datos. La información extraída de medidores inteligentes podría ser:

- Detalles del consumo de energía.
- Número de personas dentro o fuera de un edificio.
- Hábitos diarios.
- Patrón de movimiento dentro de un edificio.

Con diversas técnicas de minería de datos se puede conocer las actividades que se desarrollan en un hogar. Con esta información es fácil predecir las actividades diarias que una persona realiza en su hogar y con ello realizar un cronograma con el tiempo en el que la persona se encuentra fuera de casa para aprovechar la situación [153].

Por otro lado, se define a la privacidad como el derecho de una entidad, actuando en su propio nombre, para determinar el grado en que interactúa con su entorno, así como el grado en que la entidad está dispuesta a hacerlo [154]. La privacidad debe estar protegida en los dispositivos, en el almacenamiento durante la comunicación y en el procesamiento, lo que evitaría la divulgación de la información confidencial [155]. La privacidad de los usuarios y su protección de

datos se han identificado como uno de los desafíos más importantes que deben abordarse en dispositivos IoT. A continuación describimos las vulnerabilidades desde sus diferentes enfoques.

8.4.1. Vulnerabilidades en el Dispositivo

La información privada se puede filtrar en caso de manipulación no autorizada o manejo del *hardware* y *software*. Por ejemplo, una persona no autorizada al uso del dispositivo puede agregar configuraciones para que un dispositivo envíe información sensible no solo al servidor legítimo, sino también a él mismo o cualquier otro destino. Por lo tanto, para los dispositivos que reúnen datos sensibles, la robustez y la resistencia a la manipulación son especialmente importantes. Para garantizar la seguridad de los dispositivos IoT son necesarias la aplicación de técnicas de validación de integridad de los dispositivos, módulos resistentes a la manipulación y entornos de ejecución confiables.

Con el fin de proporcionar privacidad en los dispositivos, hay muchos problemas que deben abordarse, como podría ser la privacidad de la ubicación del dueño del dispositivo. En este sentido, la ubicación en redes de sensores inalámbricos se puede lograr mediante algoritmos como: *Multi-Routing Random walk* [156], SSRA[157], EEPR[158], entre otros.

8.4.2. Vulnerabilidades en la Comunicación

Para garantizar la confidencialidad de los datos durante la transmisión de la información, lo más común es aplicar técnicas de cifrado. Sin embargo, debido a que los dispositivos IoT tienen recursos limitados, es complicado aplicarlos a estos dispositivos. Existen iniciativas para el desarrollo de nuevas herramientas dentro de la criptografía, las cuales son algoritmos de cifrado ligero diseñados

especialmente para el IoT[159]. Por sí solas, las técnicas de cifrado son insuficientes para garantizar la confidencialidad de la información. Un correcto enfoque se puede referir a protocolos de comunicación seguros [160].

8.4.3. Vulnerabilidades en el Almacenamiento

Para proteger la privacidad en el almacenamiento de la información, se pueden considerar los siguientes principios:

- Almacenar la menor cantidad de información necesaria.
- Evitar almacenar información sensible, a no ser que sea estrictamente obligatorio.
- Minimizar el tiempo de almacenamiento de datos.
- Usar un alias para no vincular la información con una identidad real.
- Anonimizar datos.

8.4.4. Vulnerabilidades en el Procesamiento

La información personal solo debe ser usada cuando sea necesaria, con la aceptación explícita y el conocimiento del propietario de los datos. Sus datos personales no deben divulgarse ni ser retenidos por terceros. Con base en esto, es necesario mantener protegida la información recolectada a través de técnicas criptográficas [159]. Estos métodos aportan ciertas garantías de protección en caso de uso no autorizado de la información recolectada.

Investigadores de la Universidad de Pakistan recolectaron información sobre los ataques más usados en cada una de las capas del IoT, al igual que los

desafíos más comunes que se enfrentan en cada una de ellas [162], algunos de estos desafíos se incluyen en la Tabla 8.1.

La protección contra los ataques cibernéticos es uno de los principales retos a considerar en el diseño de los sistemas de IoT. Uno de los primeros ataques exitosos contra sistemas de control industrial fue el gusano Slammer, que infectó dos sistemas de monitoreo críticos de una central nuclear en los EE. UU. en el 2003 [163]. En el mismo año, un virus informático infectó el sistema de control de señal y despacho de una importante red de transporte en los EE. UU. que condujo al paro completo de trenes de pasajeros y de carga [**?**]. En los años siguientes ocurrieron muchos incidentes afectando significativamente el funcionamiento de organizaciones y sus activos[164].

También, es importante plantear como reto, mantener la disponibilidad de los sistemas, ya que, algún retraso indeseado puede significar grandes pérdidas en la productividad y eficiencia de cualquier organización o empresa.

Otro objetivo fundamental es prevenir cualquier falla del sistema que pueda resultar en daños físicos o que atenten contra la vida de un ser humano. Para lograr este objetivo, se debe preservar la integridad de los sistemas de IoT. Esto incluye la protección contra el sabotaje, que puede provocar una pérdida inadvertida de la calidad del producto. Por lo anterior, es recomendable que se tengan evidencias para demostrar a terceros que la información brindada es integra y de confianza. Mantener la integridad y la confidencialidad de cualquier elemento que utilice el IoT es un aspecto complicado, por lo tanto, siempre se debe de tener un plan estratégico que mantenga un equilibrio entre la seguridad y la disponibilidad. Si llega a ocurrir un ataque, los sistemas afectados deben ser temporalmente deshabilitados y posteriormente restaurados después del ata-

que. Sin embargo, este concepto es desechado ya que la principal tarea del IoT es la disponibilidad en cualquier momento.

Capas del IoT	**Vulnerabilidades**
Aplicación	Ataques de código malicioso, software indefenso, ataques de *phishing*.
Cómputo	Seguridad de la aplicación, seguridad de la infraestructura primaria, seguridad de datos en la computación en la nube, amenaza a los recursos compartidos, ataque a máquinas virtuales, relación de terceros.
Transmisión	Ataque de pozo, ataque de hombre en el medio, acceso autorizado RFID, ataque a la puerta de enlace, falsificación de RFID.
Percepción	Inserción de un nodo falso, inserción de código malicioso, ataque de inactividad, interruptor de nodo de red de sensor inalámbrico, ruido en datos.

Tabla 8.1: Ataques en las capas de IoT

8.5. Medidas de Prevención en el IoT

Existen mecanismos de protección de la seguridad y privacidad en dispositivos IoT. Estas medidas pueden resumirse como se muestra en la Tabla 8.2 las cuales están organizadas de acuerdo al nivel de la arquitectura del IoT [162]. Dentro de los mecanismos de protección de la privacidad en dispositivos IoT podemos encontrar esquemas de administración de privacidad que permite al usuario estimar el riesgo de compartir datos privados en medidores inteligentes [165]. También es posible encontrar soluciones basadas en la capa de transmisión de IoT, ya que se necesita autenticar ambos lados para saber que se está comunicando con la parte deseada [166]. Por otro lado, se han desarrollado esquemas donde las transmisiones de datos están protegidas por un protocolo de cifrado simétrico SHS [167].

A pesar de que el IoT es una tecnología revolucionaria, los dispositivos que se conectan a él son como cualquier otra computadora. Por lo que son igual o más vulnerables [170]. Se ha demostrado que la combinación de intentos de ataque más usados en los últimos años es "telecomadmin/admintelecom" [171]. Por ello, es necesario cambiar las contraseñas con frecuencia y asegurarse de que sean lo suficientemente robustas y fáciles de recordar. Además de comprobar regularmente los parches de seguridad y deshabilitar los dispositivos siempre y cuando no estén en uso. Algunos investigadores recomiendan instalar un *firewall* en la casa o la empresa, para restringir el acceso a usuarios no autorizados. Otro método de protección es el uso de certificaciones, es decir, permitir que los usuarios con certificados de seguridad controlen los dispositivos y a la vez se bloqueen automáticamente los demás perfiles no autorizados.

Los dispositivos conectados al IoT tienen vulnerabilidades que pueden ser explotadas fácilmente, entre ellas se encuentran las contraseñas débiles y la falta de cifrado entre las comunicaciones de los dispositivos[166]. Recientemente se han detectado ataques de malware a IoT, como Mirai, el cual es un software que desde que se hizo público su código fuente se han hecho variantes del mismo, haciendo listas de contraseñas y usuarios más largas [171]. Por otro lado, el código ocasiona que los dispositivos cercanos o conectados a la víctima sean infectados igualmente. Los gusanos, *backdoors*, *brickerbot* y *bots* de *spam*, son otro tipo de estrategias usadas para obtener información sensible. Por lo que se recomienda que se ejecuten periódicamente una exploración de puertos en todos los equipos, instalar *firewalls* y mantener todos los dispositivos actualizados. Sin embargo, a pesar de que se cuente con todas las medidas de protección posible, nunca se llega a una seguridad total. Por lo que es necesario contar con estrategias definidas para la reducción del riesgo, tomando acciones por adelantado. Adicionalmente, si es posible, se deben agregar planes de contingencia basados en acciones puestas en marcha sólo si las señales de advertencia se disparan. No se trata de cambiar la probabilidad o el impacto del riesgo, pero sí planificar como controlarlo en caso de que ocurra.

8.6. Retos de Seguridad en el IoT

A continuación se describen una serie de retos en la seguridad dentro de IoT.

8.6.1. Actualizaciones de software

Mantener los sensores de IoT correctamente calibrados es esencial para su funcionamiento, como el de cualquier otro tipo de sensor eléctrico. Los sensores de nuevas generaciones están integrados en numerosos dispositivos lo que hará difícil de sincronizar el flujo de datos de todo el *hardware* sin ayuda de un equipo profesional. También es importante considerar que todos los dispositivos cuenten con un *software* actualizado, con el fin de prevenir ataques y sean detectados y mitigados por las organizaciones que distribuyen dichos equipos.

8.6.2. Conectividad

En su forma actual IoT utiliza un modelo centralizado cliente-servidor para establecer conectividad en servidores, estaciones de trabajo y sistemas. Por el momento, dicho modelo es eficiente. De acuerdo a algunos reportes, se espera que más de 20 mil millones de dispositivos se conecten al IoT para el 2020 [173]. Es solo cuestión de tiempo para que los usuarios de IoT tengan afectaciones significativas por la gran cantidad peticiones y la capacidad de respuesta limitada a las mismas.

8.6.3. Regulaciones

Puesto que el IoT, la nube e incluso el Internet no están ligados a una ciudad, estado o alguna región específica, ¿Quién es el encargado de establecer regulaciones? Existen personas que creen que los gobiernos deberían establecer estas regulaciones, pero ¿Cómo hacerlo cuando las leyes son diferentes entre los

países? La IEEE, menciona algunas de las propuestas de los defensores de las regulaciones gubernamentales [174]. Entre ellas se destaca la importancia de la privacidad en estas regulaciones.

8.6.4. Inteligencia Artificial

Las amenazas de tipo *ransomware* han crecido 35 % en el último año. Por lo que se puede considerar que los problemas para los proveedores de la nube solo están comenzando [175][176]. Es muy probable que el siguiente objetivo del *ransomware* sean los proveedores de servicio de la nube. La caída de estos servicios pueden producir un punto de falla único para cientos de empresas, entidades gubernamentales y organizaciones de atención médica. Se prevé un incremento de *malware* creado completamente por máquinas basado en la detección de vulnerabilidades automatizada y el análisis de datos complejos aprovechado por la inteligencia artificial para crear código nuevo que sea capaz de evadir su detección.

Capas del IoT	**Medidas de prevención**
Aplicación	Validación de usuario, políticas especiales y permisos, uso de antivirus, *anti-adware* y *antispyware*, uso de firewalls, técnicas de evaluación de riesgos.
Cómputo	Cifrado para asegurar la información clasificada, dispersión de redundancia en la fragmentación de datos, bloqueo hiper seguro, aplicaciones de firewall web.
Transmisión	Confidencialidad de la información, integridad de los datos, enrutamiento seguro.
Percepción	Autenticación de dispositivos, diseño físico seguro de dispositivos finales, arranque seguro, integridad de los datos, anonimato.

Tabla 8.2: Medidas de prevención en la arquitectura IoT

Bibliografía

[1] Association for Computing Machinery. Disponible en: `https://www.acm.org/`

[2] I. Lee. (2020) "Internet of Things (IoT) Cybersecurity: Literature Review and IoT Cyber Risk Management". Frontiers in Cyber Security.

[3] M. R. Islam and K. M. Aktheruzzaman. (2020) "An Analysis of Cybersecurity Attacks against Internet of Things and Security Solutions". Journal of Computer and Communications.

[4] Rahman N. A. A, SairiI. H., ZiziN. A. M., and Khalid. F. (2020) "The Importance of Cybersecurity Education in School". International Journal of Information and Education Technology.

[5] Y. Guo, Y. Hou, Y. Hao and W. Xu. (2020) "Modeling and Analysis of Cyberspace Threat Sources Based on Vulnerabilities". International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing.

[6] A. Pawlicka, M. Choraś and M. Pawlicki. (2020) "Cyberspace threats: not only hackers and criminals. Raising the awareness of selected unusual cy-

berspace actors - cybersecurity researchers' perspective". ARES '20: Proceedings of the 15th International Conference on Availability.

[7] Bell, T., Witten, I. H. y Fellows, M. (2015). "Computer Science Unplugged", Estados Unidos.

[8] Brassard, G. y Bratley P. (1997). "Fundamentos de Algoritmia". Madrid, España: Prentice Hall.

[9] Bribiesca, E., Galaviz, J. y Rajsbaum, S. (2010). "Computación". En Enciclopedia de Conocimientos Fundamentales (509-785). Ciudad de México, México: UNAM-Siglo XXI.

[10] Cormen, T. H., Leiserson C. E., Rivest R. L. y Stein C. (2009). "Introduction to Algorithms" (2nda edición). Estados Unidos: MIT Press y McGraw-Hill.

[11] CSTA y ISTE. (2011). "Computational Thinking", Leadership Toolkit. Computer Science Teachers Association y la International Society for Technology in Education.

[12] CSTA y ISTE. (2011). "Computational Thinking", Teacher Resources. Computer Science Teachers Association y la International Society for Technology in Education.

[13] Deitel, H. M. y Deitel P. J. (2003). "Cómo Programar en C/C++", Pearson Educación.

[14] Garcia, D. (2012). "The Beauty and Joy of Computing". University of California, Berkeley. Disponible en: https://bjc.berkeley.edu/

[15] Hewitt, P. G. (2007). "Física Conceptual", México: Pearson Educación.

[16] Kramer, J. (2007). "Is abstraction the key to computing?" Communications of the ACM, 50, 36-42.

[17] Pagels, H. R. (1991). "Los sueños de la razón: el ordenador y los nuevos horizontes de las ciencias de la complejidad". España: Gedisa.

[18] Paul, R. y Elder, L.(2003). "La Mini-guía para el Pensamiento Crítico Conceptos y Herramientas". Fundación para el Pensamiento Crítico.

[19] Pitts, M. y Matson, L. (2008). "Escribir con Glifos Mayas: Nombres, lugares y oraciones simples, una introducción no técnica a los glifos mayas". The Aid and Education Project, Inc.

[20] Reynolds, C. W. (1987). "Flocks, herds and schools: A distributed behavioral model". SIGGRAPH '87: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, 21, 25-34.

[21] Starir, R. M. y Reynolds, G. W. (2000). "Principios de Sistemas de Información". International Thomson Editores.

[22] Tanenbaum, A. S. (2003). "Redes de Computadoras". Pearson Educación.

[23] Wing, J. M. (2008). "Computational Thinking and Thinking about Computing". Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 366, 3717-3725.

[24] Wing, J. M. (2006). "Computational Thinking". Communications of the ACM, 49, 33-35.

[25] Young, H. D. y Roger A. F. (2009), "Física Universitaria", Volumen 1, México: Pearson Educación.

[26] Zapotecatl, J. L., Munoz-Meléndez A. y Gershenson C. (2016). "Performance Metrics of Collective Coordinated Motion in Flocks". Proceedings of the Artificial Life Conference 2016, 322–329.

[27] Gobierno de México. (1917). "Constitución Política de los Estados Unidos Mexicanos".

[28] Gobierno de México. (2010). "Ley Federal de Protección de Datos Personales en Posesión de Particulares".

[29] Gobierno de México. (2011). "Reglamento de la Ley Federal de Protección de Datos Personales en Posesión de los Particulares".

[30] Organización de Estados Americanos. (1969). "Convención Americana sobre Derechos Humanos" (Pacto de San José).

[31] Naciones Unidas. (2004). "Pacto Internacional de Derechos Civiles y Políticos".

[32] Suprema Corte de Justicia de la Nación en México. (2014). Gaceta del Semanario Judicial de la Federación, Libro 5, Abril de 2014, Tomo I, página 96.

[33] Gobierno de México. (2016). "Ley Federal de Transparencia y Acceso a la Información Pública Gubernamental". DOF 27-01-2017.

[34] Gobierno de México. (1889). "Código de Comercio". DOF-30-12-2019.

[35] Gobierno de México. (1981). "Código Fiscal de la Federación". DOF 09-01-2020.

[36] Gobierno de México. (2012). "Ley de Firma Electrónica Avanzada". DOF 11-01-2012.

[37] Gobierno de México. (2012). "Decreto de Austeridad". DOF 10-12-2012.

[38] "Parlamento Europeo y Consejo de la Unión Europea". (2016). Reglamento General de Protección de Datos. Relgamento (UE) 2016/679.

[39] ISO/IEC. (2013). "Information security management".

[40] ISO. (2018). "Risk Management".

[41] Secretaría de Hacienda y Crédito Público. (2020). "Resolución Miscelánea Fiscal vigente".

[42] Menezes, A., van Oorschot, P., Vanstone, S., Rosen, K. (1997). "Handbook of Applied Cryptography". Boca Raton: CRC Press.

[43] Singh, S., (2000). "The Code Book: The science of secrecy from ancient Egypt to quantum cryptography". New York: Anchor.

[44] Yaschenko, V. V., (2009). "Cryptography: An introduction". Orient black swan.

[45] National Institute of Standards and Technology. (2018). "Post-quantum cryptography PQC". `https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization`

[46] Cardinal, D., (2019). "Quantum cryptography demystified: How it works in plain language". Extreme tech, 11 March.

[47] Shor, P. W. (1994). "Algorithms for quantum computation: Discrete logarithms and factoring". 35th Annual symposium on foundations of computer science, 1994 Proceedings., November. IEEE

[48] Bernstein, D. J. (2009). "Introduction to post-quantum cryptography". Springer, Berlin, Heidelberg.

[49] Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional. Disponible en: `https://www.cinvestav.mx/`

[50] Instituto Politécnico Nacional. Disponible en: `https://www.ipn.mx/`

[51] Universidad Iberoamericana. Disponible en: `https://ibero.mx/`

[52] Universidad La Salle. Disponible en: `https://lasalle.mx/`

[53] Universidad Tecnológica de México. Disponible en: `https://www.unitec.mx/campus-sur/`

[54] Instituto Nacional de Astrofísica, Óptica y Electrónica. Disponible en: `https://www.inaoep.mx/`

[55] Tecnológico de Monterrey. Disponible en: `https://maestriasydiplomados.tec.mx/posgrados/maestria-en-ciberseguridad?locale=es-MX`

[56] Universidad Autónoma de Nuevo León. Disponible en: `https://www.uanl.mx/`

[57] Universidad en Internet. Disponible en: `https://www.unir.net/`

[58] Centro de Estudios Superiores Navales. Disponible en: `https://cesnav.uninav.edu.mx/cesnav/index_inicio.html`

[59] Universidad Nacional Autónoma de México. Disponible en: `https://www.unam.mx/`

[60] Universidad del Valle de México. Disponible en: `https://uvm.mx/la-uvm/campus/campus-coyoacan`

[61] Universidad Anahuac. Disponible en: `https://online.anahuac.mx/diplomado-en-linea-ciberseguridad.html`

[62] Matt Blaze, Whitfield Diffie, Ronald L. Rivest, Bruce Schneier, Tsutomu Shinomura, Eric Thompson, and Michael Wiener. (1996) "Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security". Released by the Business Software Alliance (January 1996). (1996-01)

[63] Daniel J. Bernstein (2007). "The Salsa20 family of stream ciphers". (2007-12-24).

[64] Walter Tuchman (1997). "A brief history of the data encryption standard". Internet besieged: countering cyberspace scofflaws. ACM Press/Addison-Wesley Publishing Co. New York, NY, USA. pp. 275–280.

[65] Xuejia Lai and James L. Massey, (1990) "A Proposal for a New Block Encryption Standard", EUROCRYPT 1990, pp. 389–404.

[66] Daemen, Joan; Rijmen, Vincent (2003) "AES Proposal: Rijndael". (March 9, 2003).

[67] Bruce Schneier. "Cryptography: Description of a New Variable-Length Key", 64-Bit Block Cipher (Blowfish) - Schneier on Security.

[68] Smart, N. P., Paterson, K., and Cramer, R. (2016). "Cryptography made simple", volume 53.

[69] Paar, Christof y Pelzl, J. (2013). "Understanding cryptography", volume 53.

[70] Shor, P. (1997) "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer", SIAM Journal Computing, Vol. 26, No. 5, pp. 1484–1509.

[71] Katz, J. (2007). "Introduction to Modern Cryptography".

[72] Dang, Q. H. (2012). Recommendation for applications using approved hash algorithms. https://doi.org/10.6028/NIST.SP.800-107r1

[73] SP 800-107 Rev. 1, Recommendation for Applications Using Approved Hash Algorithms | CSRC. (n.d.)

[74] L. Lamport, "Password Authentication with Insecure Communication", Communications of the ACM 24.11 (November 1981), pp 770-772.

[75] Merkle, R. C. (1988) "A Digital Signature Based on a Conventional Encryption Function". Advances in Cryptology — CRYPTO '87. Lecture Notes in Computer Science. 293. pp. 369–378. ISBN 978-3-540-18796-7.

[76] Koblitz, Neal; Menezes, Alfred J. (January 2016). "Cryptocash, cryptocurrencies, and cryptocontracts". Designs, Codes and Cryptography. 78 (1): 87–102.

[77] "General Verifiable Federation". Google Wave Protocol.

[78] NIST (2013) "FIPS PUB 186-4: Digital Signature Standard"

[79] Diffie, W; M. E. Hellman. (1976) "New Directions in Cryptography", IEEE Transactions On Information Theory, Vol. IT-22, No. 6, pp. 644-654

[80] Rivest, R. L.; A. Shamir; L. Adleman. (1978) "A method for obtaining digital signatures and public-key cryptosystems", Communications of the ACM, Vol. 21, No. 2, pp. 120-126,

[81] ACM A.M. Turing Award Winners by Year. `https://amturing.acm.org/byyear.cfm`

[82] Zimmermann, P., et al. "Factorization of RSA-250". Disponible en: `https://lists.gforge.inria.fr/pipermail/cado-nfs-discuss/2020-February/001166.html`

[83] NIST (2020). "NIST SP 800-57. Recommendations for Key Management – Part 1: General". Rev. 5

[84] ITU-T (2008). "Recommendation X.509". Information technology – Open systems interconnection – The Directory: Public-key and attribute certificate frameworks

[85] NIST (2020) "NISTIR 8309. Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process".

[86] F. Boudot, P. Gaudry, A. Guillevic, N. Heninger, E. Thomé, and P. Zimmer-man, "Factorization of rsa-250," February

2020.https://lists.gforge.inria.fr/pipermail/cado-nfs-discuss/2020-February/001166.html.

[87] E. B. Barker, "Recommendation for key management: Part 1 -general, rev 5," May 2020. `https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5-draft.pdf`.

[88] International Telecommunication Union, "Information technology – opensystems interconnection – the directory: Public-key and attribute certificateframeworks". ITU-T Recommendation X.509, October 2019.

[89] D. Moody, G. Alagic, D. C. Apon, D. A. Cooper, Q. H. Dang, J. M. Kelsey,Y.-K. Liu, C. A. Miller, R. C. Peralta, R. A. Perlner, A. Y. Robinson, D. C.Smith-Tone, and J. Alperin-Sheriff, "Status report on the second round ofthe nist post-quantum cryptography standardization process." NIST Inter-agency/Internal Report (NISTIR) - 8309, October 2020.

[90] P. S. L. M. Barreto.(2009) "Introduction to syndrome-decoding-based cryptography". Invited talk. CryptoSeminar DCU 2009 series. http://www.computing.dcu.ie/ ldominguez/index.php?n=Seminar.SyndromeDecoding.

[91] P. S. L. M. Barreto. (2010) "Post-Quantum Cryptography: Introduction and Trends". Invited talk at the JQI/NIST/UMD Quantum Information Workshop. Disponible: http://www.nature.com/news/2010/100829/full/news.2010.436.html

[92] P. S. L. M. Barreto. (2009) "Post-quantum cryptosystems based on coding theory: overview and recent developments". Western

European Workshop on Research in Cryptology. Invited talk. http://2009.weworc.org/files/baretto.pdf.

[93] D. Bernstein, J. Buchmann, and E. Dahmen, editors. (2009) "Post-Quantum cryptography". Springer Berlin / Heidelberg.

[94] N. Courtois, M. Finiasz, and N. Sendrier. (2001) "How to achieve a mceliece-based digital signature scheme". In ASIACRYPT, pages 157–174.

[95] O. Goldreich, S. Goldwasser, and S. Halevi. (1997) "Public-key cryptosystems from lattice reduction problems". In CRYPTO, pages 112–131.

[96] L. K. Grover. (1996) "A fast quantum mechanical algorithm for database search". In Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing (STOC), pages 212–219. ACM.

[97] J. Hoffstein, J. Pipher, and J. H. Silverman. (1998) "Ntru: A ring-based public key cryptosystem". In ANTS, pages 267–288.

[98] C. W. D. E. J. S. L. Lydersen, C. Wiechers and V. Makarov. (2010) "Hacking commercial quantum cryptography systems by tailored bright illumination". Nature Photonics, 4:686–689.

[99] A. K. Lenstra, H. W. Lenstra, and L. Lovasz. (1982) "Factoring polynomials with rational coefficients". Mathematische Annalen, 261:515–534, 1982.10.1007/BF01457454.

[100] R. J. McEliece. (1978) "A Public-Key Cryptosystem Based On Algebraic Coding Theory". Deep Space Network Progress Report, 44:114–116.

[101] Z. Merali. (2010) "Hackers blind quantum cryptographers". http://www.nature.com/news/2010/100829/full/news.2010.436.html.

[102] D. Micciancio. (1998) "On the Hardness of the Shortest Vector Problem". PhD thesis.

[103] H. Niederreiter. (1986) "Knapsack-type cryptosystems and algebraic coding theory". Problems Control Inform. Theory/Problemy Upravlen. Teor. Inform., 15(2):159–166.

[104] Cunha, F., Villas, L., Boukerche, A., Maia, G., Viana, A., Mini, R. A.,and Loureiro, A. A. (2016). Data communication in vanets: Protocols, applications and challenges.Ad Hoc Networks, 44:90 – 103.

[105] K. Kobayashi, Y. Totani, K. Utsu, and H. Ishii. Achieving secure communication over manet using secret sharing schemes. The Journal of Supercomputing,72(3):1215–1225, Mar 2016.

[106] K. Plößl and H. Federrath. A privacy aware and efficient security infrastructure for vehicular ad hoc networks. Computer Standards Interfaces, 30(6):390 – 397,2008. Special Issue: State of standards in the information systems security area.

[107] S. Busanelli, G. Ferrari, and L. Veltri. Short-lived key management for secure communications in vanets. In 2011 11th International Conference on ITS Telecommunications, pages 613–618, Aug 2011.

[108] X. Liu, Z. Fang, and L. Shi. Securing vehicular ad hoc networks. In2007 2ndInternational Conference on Pervasive Computing and Applications, pages 424–429, July 2007.

[109] M. N. Mejri, J. Ben-Othman, and M. Hamdi. Survey on vanet security challenges and possible cryptographic solutions. Vehicular Communications, 1(2):53 – 66,2014.

[110] N.-W. Wang, Y.-M. Huang, and W.-M. Chen. A novel secure communication scheme in vehicular ad hoc networks. Comput. Commun., 31(12):2827–2837, July 2008.

[111] N. Chaubey. Security analysis of vehicular adhoc networks (VANETS): A comprehensive study. (5):261–273, 07 2016.

[112] S. R. Ausekar and S. K. Pasupuleti. Dynamic verifiable outsourced database withfreshness in cloud computing.Procedia computer science, 143:367–377, 2018.

[113] M. Etemad and A. Küpçü. Verifiable database outsourcing supporting join.Journal of Network and Computer Applications, 115:1 – 19, 2018.

[114] L. Ferretti, M. Marchetti, M. Andreolini, and M. Colajanni. A symmetric cry-ptographic scheme for data integrity verfication in cloud databases.InformationSciences, 422:497 – 515, 2018.

[115] E. Narashima, M. mykletun, and G. Tsudik.Signature bouquets: Immutabi-lity for aggregated/condensed signatures. InEuropean Symposium on Researchin Computer Security, pages 160–176. Springer, 2004.

[116] L. M. Rodríguez-Henríquez and D. Chakraborty. RDAS: A symmetric key sche-me for authenticated query processing in outsourced databases. In R. Accorsi andS. Ranise, editors,Security and Trust Management, pages 115–130, Berlin, Heidel-berg, 2013. Springer Berlin Heidelberg

[117] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes.In J. Stern, editor,Advances in Cryptology — EUROCRYPT '99, pages 223–238,Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[118] J. D. i Ferrer. A new privacy homomorphism and applications.Information Pro-cessing Letters, 60(5):277 – 282, 1996.

[119] M. Will.A guide to homomorphic encryption, pages 101–127. 12 2015.

[120] J. Dyer, M. Dyer, and K. Djemame. Order-preserving encryption using appro-ximate common divisors.Journal of Information Security and Applications, 49,Dec. 2019.

[121] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Order preserving encryption fornumeric data. InSIGMOD '04: Proceedings of the 2004 ACM SIGMOD interna-tional conference on Management of data, pages 563–574, New York, NY, USA,2004. ACM Press.

[122] A. Boldyreva, N. Chenette, and A. O'Neill. Order-preserving encryption revisi-ted: Improved security analysis and alternative solutions. pages 578–595, 08 2011.

[123] R. A. Popa, F. H. Li, and N. Zeldovich. An ideal-security protocol for order-preserving encoding. In2013 IEEE Symposium on Security and Privacy, pages 463–477, 2013.

[124] M. Bellare, T. Ristenpart, P. Rogaway, and T. Stegers. Format-preserving encry-ption.IACR Cryptology ePrint Archive, 2009:251, 2009.

[125] M. Weiss, B. Rozenberg, and M. Barham.Practical solutions for format-preserving encryption.CoRR, abs/1506.04113, 2015.

[126] Lawrence C. Paulson. (1998) "The inductive approach to verifying cryptographicprotocols".Journal in Computer Security, 6(1-2):85–128.

[127] M. Abadi and P. Rogaway. (2002) "Reconciling two views of cryptography (the computational soundness of formal encryption)". Journal of Cryptology,15(2):103–127.

[128] Anupam Datta, Ante Derek, John C. Mitchell, Vitaly Shmatikov, and Mathieu Turuani. (2005) "Probabilistic polynomial-time semantics for a protocol security logic". InICALP, pages 16–29.

[129] B. Pfitzmann, M. Schunter, and M. Waidner. (2000) "Cryptographic security of reactive systems". Electronic Notes in Theoretical Computer Science(ENTCS), 32:1–19.

[130] M. Backes, B. Pfitzmann, and M. Waidner. (2003) "A universally composable cryptographic library".

[131] Jonathan Herzog. (2004) "Computational Soundness for Standard Assumptions of Formal Cryptography". PhD thesis, Massachusetts Institute of Technology.

[132] Jonathan C. Herzog. (2003) "The Diffe-Hellman key-agreement scheme in the strand-space model". In CSFW, pages 234–247.

[133] M. Baudet, V. Cortier, and S. Kremer. "Computationally sound implementations of equational theories against passive adversaries". In Proceedings of the 32nd International Colloquium on Automata, Languages and Pro-gramming (ICALP'05), volume 3580 ofLecture Notes in Computer Science,pages 652–663. Springer, 2005.

[134] Steve Kremer and Laurent Mazaré. (2007) "Adaptive soundness of static equivalence". In Joachim Biskup and Javier Lopez, editors, Proceedings of the 12th European Symposium on Research in Computer Security (ESORICS'07), volume 4734 of Lecture Notes in Computer Science, pages 610–625. Springer.

[135] Juan Carlos López-Pimentel. (2008) "On the Automated Correction of Faulty Security Protocols". (PhD thesis). PhD thesis, Tecnológico de Monterrey, Campus Estado de México.

[136] Gavin Lowe. (1997) "A hierarchy of authentication specifications". In CSFW '97: Proceedings of the 10th Computer Security Foundations Workshop, page 31, Rockport, Massachusetts, USA. IEEE Computer Society.

[137] D. Dolev and A. C. Yao. (1983) "On the security of public key protocols". Technical Report 2, Stanford University, Stanford, CA, USA.

[138] Gavin Lowe. (1996) "Breaking and fixing the Needham-Schroeder public-key protocol using FDR". In TACAs '96: Proceedings of the Second International Workshop on Tools and Algorithms for Construction and Analysis of Systems, pages 147–166, London, UK. Springer-Verlag.

[139] M. Burrows, M. Abadi, and R. M. Needham. (1989) "A logic of authentication". Proceedings of the Royal Society of London, 426(1):233–71.

[140] Stephen H. Brackin. (1996) "A HOL extension of GNY for automatically analyzing cryptographic protocols". In CSFW'96: Proceedings of The 9th Computer Security Foundations Workshop, page 62, Washington, DC, USA, 1996.IEEE Computer Society Press.

[141] Catherine Meadows. (1996) "The NRL protocol analyzer: An overview". Journal of Logic Programming, 26(2):113–131.

[142] AVISPA Team. (2005) AVISPA v1.0 User Manual, v1.0 edition, June.

[143] F. J. Thayer Fabrega, J.C. Herzog, and J.D. Guttman. (1998) "Strand spaces: Why is a security protocol correct?". In Proceedings Symposium on Security and Privacy. volume 15, pages 160–171, Oakland, CA, USA. IEEE computer Society.

[144] X. D. Song, S. Berezin, and A. Perrig. (2001) "Athena: A novel approach to efficient automatic security protocol analysis". Journal of Computer Security, 9(1-2):47–74.

[145] G. Steel, A. Bundy, and M. Maidl. (2003) "Attacking the Asokan-Ginzboorg pro-tocol for key distribution in an ad-hoc bluetooth network using coral". In H. König, M. Heiner, and A. Wolisz, editors, IFIP TC6 /WG 6.1: Pro-ceedings of 23rd IFIP International Conference on Formal Techniques for Networked and Distributed Systems, volume 2767, pages 1–10, Berlin, Ger-many, 2003. FORTE 2003 (work in progress papers).

[146] M. Abadi and R. Needham. (1996) "Prudent engineering practice for cryptographic protocols". IEEE Transactions on Software Engineering, 22(1):6–15.

[147] Escamilla-Ambrosio PJ, Salinas-Rosales M, Acosta-Bermejo R. (2013) "Internet de las cosas: estado actual, retos y perspectivas". Memorias 1er Congreso Iberoamericano de Instrumentación y Ciencias Aplicadas - SOMI XXVIII Congreso de Instrumentación. Sn. Francisco de Campeche, Campeche, México, 28-31 de octubre.

[148] Centro de Noticias ONU. "La población mundial alcanza hoy 7.000 millones".

[149] Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025. Disponible en: `http://www.un.org/spanish/News/story.asp?newsID=22135#.UiZRHRvTvNI`

[150] Mohammed R.A., Djamel T., Imed R. (2015) "Architecting the Internet of Things: State of the Art". Springer, Cham: Robots and Sensor Clouds, Vol 36, pp. 55-75, August 2015.

[151] Costas E., Georgios K. (2010) "Smart Grid Privacy via Anonymization of Smart Metering Data". IEEE:2010 First IEEE International Conference on Smart Grid Communications, November 2010.

[152] Georgios K., Costas E., Stojan Z.D., Tim A.L, Rafael C. (2010) "Privacy for Smart Meters: Towards Undetectable Appliance Load Signatures". IEEE: 2010 First IEEE International Conference on Smart Grid Communications, October 2010.

[153] Michael D., Barry P.H. (2018) "Non-Intrusive Load Monitoring using Electricity Smart Meter Data: A Deep Learning Approach". Research Gate:University College Cork, November 2018.

[154] R. Shirey. (2000) "Internet security and privacy". BBN Technologies, Network Working Group, May 2000.

[155] Samiksha R.S. (2016) "A Study on Privacy and Security concerns in Internet of Things". International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, Issue 9, September 2016.

[156] Liming Z., Qiaoyan W,. Hua Z. (2012) "Preserving Sensor Location Privacy in Internet of Things". IEEE:2012 Fourth International Conference on Computational and Information Sciences, pp. 856-859, September 2012.

[157] Sofiane H., Jaime L., Pascal L. (2018) "Smart and Self-Organized Routing Algorithm for Efficient IoT communications in Smart Cities". IET Wireless Sensor Systems, October 2018.

[158] Sang H.P., Seungryong C. and Jung R. L., (2014) "Energy-Efficient Probabilistic Routing Algorithm for Internet of Things". Hindawi Publishing Corporation, Journal of Applied Mathematics, Volume 2014, April 2014.

[159] Saurabh S., Pradip K. S., Seo Y. M., Jong H. P. (2017) "Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions". Springer Berlin Heidelberg:Journal of Ambient Intelligence and Humanized Computing, pp. 1–18, May 2017.

[160] Michalis G., Korina K., Nikos F., Giannis F.M., George C.P. (2013) "Towards Secure and Context-Aware InformationLookup for the Internet of Things". IEEE International Conference on Computing, Networking and Communications (ICNC), January.

[161] Aruba HP. "El internet de las cosas IoT". [Online] Available:`https://www.arubanetworks.com/es/partners/partners-de-ecosistema/el-internet-de-las-cosas-iot/`

[162] Tariq A.R., Ehsan-ul-Haq. (2018) "Security Challenges Facing IoT Layers and its Protective Measures". University of Pakistan Lahore, Pakistan, Vol. 179, Issue. 27, March 2018.

[163] Kevin P., SecurityFocus. (2003) "Slammer worm crashed Ohio nuke plant network". SecurityFocus, August 2003.

[164] Ahmad-Reza S., Christian W., Michael W. "Security and Privacy Challenges in Industrial Internet of Things". IEEE:2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC), July 2015.

[165] Arijit U., Soma B., Arpan P. (2014) "IoT-Privacy: To be private or not to be private". IEEE: 2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 123-124, July 2014.

[166] Gergely A., Lejla B., Lynn B., Veelasha M., Anna K., Antoine G., Iynkaran N. (2016) "New Directions in IoT Privacy Using Attribute-Based". Radboud Repository of the Radboud University Nijmegen, pp. 463-464, May 2016.

[167] Tianyi S., Ruinian L., Bo M., Jiguo Y., Xiaoshuang X., Xiuzhen C. (2017) "A Privacy Preserving Communication Protocol for IoT Applications in Smart Homes". IEEE:IEEE Internet of Things Journal, Vol. 4, Issue. 6, pp. 1844-1852, December 2017.

[168] Alexander K., Oleg K., Kirill I. (2017) "DDoS attacks in Q3 2017". SecureList, November 2017.

[169] Statista (2017) "Industries most frequently targeted by denial of service (DDoS) attacks worldwide as of 4th quarter".

[170] Ebraheim A., Abdallah T. (2015) "Internet of Things: Features, Challenges, and Vulnerabilities". International Journal of Advanced Computer Science and Information Technology (IJACSIT). Vol. 4, Issue. 1, pp. 1-13.

[171] Constantinos K., Georgios K., Angelos S., Jeffrey V. (2017) "DDoS in the IoT: Mirai and Other Botnets". IEEE:Browse Journals & Magazines, Vol. 50, Issue 7, pp 80-84.

[172] Vala A. (2018) "New Research Uncovers Big Shifts In Costumer Expectations And Trust". Salesforce, Jun 2018.

[173] Muhammad A., Peter T., Shahid M., Jonathan R. (2017) "Context-aware cooperative testbed for energy analysis in beyond 4G networks". Springer:Telecommunication Systems, Vol. 64, Issue 2, pp 225–244, February 2017.

[174] IEEE. (2018) "Should the Government Regulate IoT Devices?". IEEE Inovation at work.

[175] Ibrar Y., Ejaz A., Muhammad H., Abdelmuttlib I., Abdalla A., Mohammed A., Muhammad., Mohsen G. (2017) "The rise of ransomware and emerging security challenges in the Internet of Things". Elsevier:Computer Networks, Vol. 129, Part 2, pp. 444-458, December 2017.

[176] Amin A., Ali D., Mauro C., Kim-Kwang R. (2018) "Detecting crypto-ransomware in IoT networks based on energy consumption footprint".

Springer Berlin Heidelberg:Journal of Ambient Intelligence and Humanized Computing, Vol. 9, Issue 4, pp. 1141–1152, August 2018.

Introducción a la Ciberseguridad y sus aplicaciones en México,
se terminó el 30 de noviembre del 2020. A partir del 1 de enero de 2021 está disponible en
formato PDF en la página de la Academia Mexicana de Computación
`http://www.amexcomp.mx`