Protocolos y aplicaciones criptográficas

75.06 Organización de Datos – Cátedra Argerich
Facultad de Ingeniería
Universidad de Buenos Aires

Objetivos de la criptografía

- **Confidencialidad**: Mantener en secreto un mensaje a la vista de ojos no autorizados
- **Autenticación**: Certificar que quien envió el mensaje es quien dice ser.
- **Integridad**: Acreditar que el mensaje no fue alterado total o parcialmente en el camino.
- **No repudio**: Impedir que quien envió el mensaje falsamente niegue su autoría.

Protocolo

- Un protocolo es una serie de pasos, que involucra a dos o más partes, designados para resolver una tarea.

Características de los protocolos criptográficos

- Debe ser completo.
- No debe ser ambiguo.
- Debe ser conocido por las partes.
- Tiene que ser acatado y cumplido.
- No debe ser posible hacer más o conocer más de lo que dice el protocolo.

Tipos de protocolos

- Protocolos arbitrados (árbitro)
- Protocolos adjudicados (juez)
- Protocolos autosuficientes

Comunicaciones con criptosistemas simétricos

1. Alice y Bob se ponen de acuerdo sobre el criptosistema a utilizar
2. Alice y Bob acuerdan una clave
3. Alice encripta el mensaje usando la clave y lo envía a Bob
4. Bob desencripta el mensaje con la clave y lee el mensaje

- Para el intercambio de claves se puede utilizar el metodo de intercambio de claves Diffie-Hellman…
- … que es vulnerable al ataque del "hombre en el medio" (**Man-in-the-middle attack)**

Comunicaciones con criptosistemas asimétricos

1. Alice y Bob se ponen de acuerdo sobre el criptosistema a utilizar
2. Bob envía a Alice su clave pública
3. Alice encripta el mensaje con la clave pública de Bob y se lo envía
4. Bob desencripta el mensaje con su clave privada y lo lee

- La comunicación mediante encriptación asimétrica es en general lenta y tarda en encriptar y desencriptar

Comunicaciones con criptosistemas híbridos

1. Bob envía a Alice su clave pública
2. Alice genera una clave temporal, la encripta utilizando la clave pública de Bob y se la envía
3. Bob desencripta la clave temporal
4. Ambos encriptan sus mensajes durante la sesión utilizando la clave temporal

Qué es una firma?

- Su fin es **identificar, asegurar o autentificar** la identidad de un autor o remitente
- Se utiliza como una **prueba del consentimiento** y/o de **verificación de la integridad** y aprobación de la información contenida en un documento o similar
- Tiene carácter legal.

Firma digitales – Caracteristicas

- Únicas
- Infalsificables
- Verificables
- Innegables
- Viables

Firma digitales

- Aplicación criptográfica que permite al receptor de un mensaje firmado digitalmente determinar la entidad originadora de dicho mensaje (**autenticación de origen y no repudio**), y confirmar que el mensaje no ha sido alterado desde que fue firmado por el originador (**integridad**).

Firma digital con criptosistemas simétricos

1. **Alice** encripta su mensaje para **Bob** con una clave Ka y la envía a **Trent**
2. **Trent** desencripta el mensaje con Ka
3. **Trent** une el mensaje a una certificación de que recibió el mensaje de parte de **Alice**, encripta todo esto con Kb, y se lo envía a **Bob**
4. **Bob** desencripta lo recibido con Kb. Lee el mensaje y la certificación de **Trent**

Firma digital con criptosistemas asimétricos

- Requiere que se pueda revertir mediante la clave publica un mensaje encriptado con la clave privada.

1. Alice firma (encripta) el mensaje con su clave privada
2. Alice envía el mensaje a Bob
3. Bob desencripta el mensaje con la clave pública de Alice, verificando la firma

Message digest

- Se utiliza una función de háshing criptográfica que toma una longitud arbitraria de texto plano y con el computa una cadena de longitud fija de bits.
- Propiedades:
  - Dado un P, es fácil calcular MD(P)
  - Dado un MD(P) es efectivamente imposible encontrar P
  - Dado un P nadie puede encontrar un P' tal que MD(P') = MD(P)
  - El cambio de al menos un bit en el input produce una salida totalmente diferente.

Firma digital con message digest

1. Alice y Bob se ponen de acuerdo sobre el criptosistema y función de Hash a utilizar.
2. Alice procesa el mensaje con la función de Hash (MD)
3. Alice firma (con su clave privada) el resultado del MD
4. Alice envia a BoB el mensaje y la firma
5. Bob procesa el mensaje recibido con la función de Hash (MD)
6. Bob encripta la firma recibida con la clave publica de Alice
7. Bob verifica que sean iguales el MD del mensaje recibido y el obtenido luego de procesar la firma.

Bit-Commitment

- Esquema que permite a una persona **elegir un valor** manteniendolo en **secreto** a los demas
- Debe permitir revelar el resultado pasado un hito temporal establecido sin permitir modificarlo.

- Utilizado como base de varias aplicaciones criptográficas.

Bit-commitment con criptosistema simétrico

1. Bob Genera un una cadena aleatoria de bits “R” y se la envía a Alice
2. Alice crea un mensaje con la cadena de bits “R” y su pronóstico (o compromiso). Encripta el resultado con una clave “K” y envía el resultado a Bob.

   Verificación:

3. Alice envía la clave privada “K”
4. Bob desencripta el mensaje para ver la predicción. Y chequea la cadena “R” para ver si es la misma que el mando en primer lugar.

Fair Coin Flips

1. Alice selecciona un bit aleatorio (cara o seca) y utiliza el protocolo de bit-commitment para protegerlo
2. Bob trata de adivinar el resultado
3. Alice revela el resultado a Bob (resolviendo el protoco bit-commitment). Bob gana si acertó.

Firma a ciegas

- Esquema que permite a una persona **firmar un mensaje** sin conocer el contenido del mismo.

- Se requiere:
  - Una función de Firma S' conocida solo por el firmante y la correspondiente función pública S que satisfaga que $S(S'(x))=x$
  - Una función conmutativa C y su inversa C' ambas conocidas solo por el emisor que satisfaga $C'(S'(C(x)))=S'(x)$

- Utilizado como base de varias aplicaciones criptográficas.

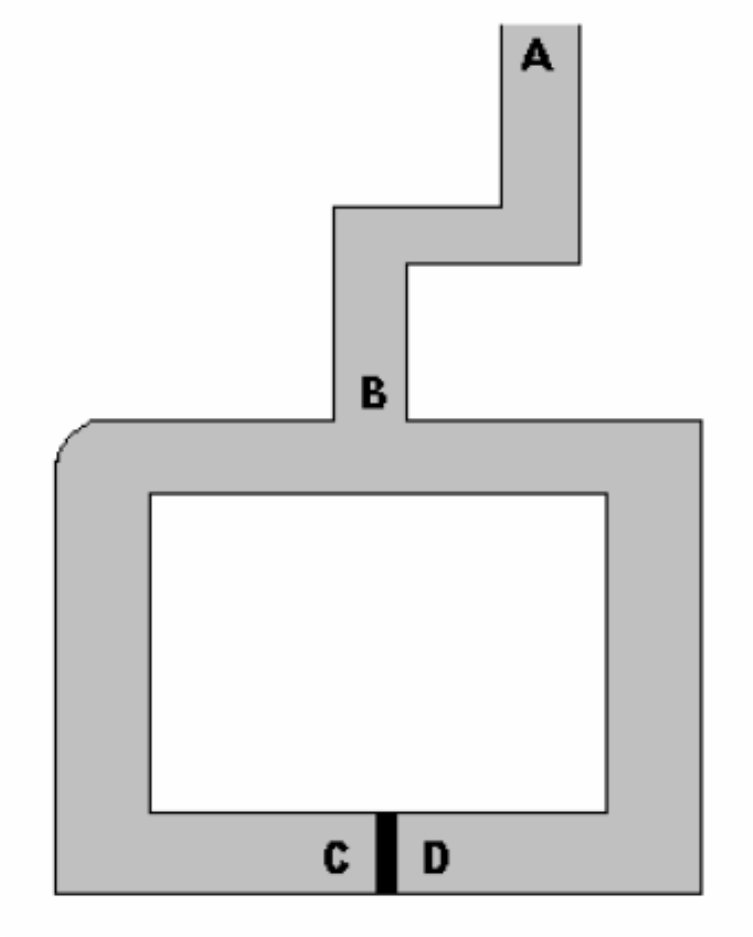
Firma a ciega con criptosistema asimetriro (RSA)

- Alice encripta con su clave publica un mensaje y se lo envia a Trent
- Trent recibe el mensaje, lo firma con su clave privada y se lo envia a Alice
- Alice recibe el mensaje y lo desencripta con su clave privada quedando así el mensaje original firmado por Trent

Zero-knowledge proofs

- Su objetivo es probar que una declaración es cierta, sin revelar nada más que la veracidad de la declaración
- Es un método iterativo.
- Debe satisfacer 3 propiedades:
  - **Totalidad**: si la afirmación es verdadera, un observador honesto queda convencido del conocimiento.
  - **Solvencia**: si la declaración es falsa el declarante deshonesto no podrá convencer (excepto con alguna probabilidad pequeña).
  - **Conocimiento cero**: si la afirmación es verdadera, un observador engañoso no aprende otra cosa mas que este hecho.

El problema de la cueva



1. Bob que no conoce la clave se ubica en el punto “A”
2. Alice (que conoce la clave) entra en la cueva y va hasta el punto “C” o el “D”
3. Bob se dirige al punto “B”
4. Bob le pide a Alice que salga:
   - Por el lado derecho o
   - Por el lado izquierdo
5. Alice cumple lo pedido utilizando la combinación para pasar por la puerta (si es necesario)
6. Los pasos 1 a 5 se repiten “n” veces.

Voto electrónico

Proceso de votacion:

- Cada votante genera N set de mensajes conteniendo el set de todos las opciones a elegir + un identificador random
- Cada votante utiliza el protocolo de firma a ciegas + zero knowledge proof con cada set y se lo envia al verificador
- El verificador abre N-1 set de mensajes y verifica que el votante no haya votado previamente y que este bien formado los sets. Luego firma cada mensaje del set no desencriptado, se los envia al votante y registra que el votante ya realizó este paso.
- El votante desencripta los mensajes del set restante
- El votante selecciona la opcion deseada + firma del verificador + identificado y la encripta con la clave publica del verificadfor y envia el voto.

Finalizada la votacion el verificador :

- desencripta los votos con su clave privada,
- verifica su firma,
- controla que el identiticador no este duplicado
- cuentra el voto
- agrega el identificador para controles de duplicados.

Digital Cash

Características deseables:

- Transferencia: El Digital Cash debe poder ser transferido a otro usuario.
- Seguridad: No debe poder ser copiado y reusado.
- Privacidad: No debe poder ser rastreado las relaciones entre un usuario y sus compras.
- Divisibilidad: Una pieza de Digital Cash de un monto dado debe poder ser subdividido en piezas de montos menores.
- Independencia: La seguridad del efectivo digital no debe ser dependiente de ninguna locación física. Este deberá poder ser transferido por redes de computadoras.
- Pago Off-line: Pagando con Digital Cash no es necesario estar enlazado a un host para procesar el pago

Digital Cash – ejemplo básico

1. Alice prepara 100 órdenes anónimas de dinero por $1000 c/u
2. Alice pone en cada una de ellas junto con un pedazo de papel carbónico en 100 sobres diferentes. Entrega los mismos al banco
3. El banco abre 99 sobres y confirma que cada orden es por $1000
4. El banco firma el sobre que quedó cerrado. La firma pasa del sobre mediante el papel carbónico a la orden de compra. El banco le retorna el sobre a Alice y descuenta $1000 de su cuenta.
5. Alice abre el sobre y gasta la orden de compra en un comercio
6. El comerciante verifica la firma del banco para verificar que la orden sea legitima.
7. El comerciante lleva la orden al banco
8. El banco verifica su firma y acredita $1000 en la cuenta del comerciante.

Bitcoins

- Red consensuada y una moneda completamente digital
- Sin una autoridad central o intermediarios
- Pago descentralizado impulsado por sus usuarios
- La primera especificación del protocolo Bitcoin y la prueba del concepto la publicó Satoshi Nakamoto en el 2009 en una lista de correo electrónico

bitcoin

- La unidad monetaria se llama “bitcoin”
- Es totalmente digital
- Pueden existir un máximo de 21 millones de bitcoins
- Puede dividirse hasta en 8 decimales

Direcciones – wallets

- Para participar en la red se debe tener una dirección.

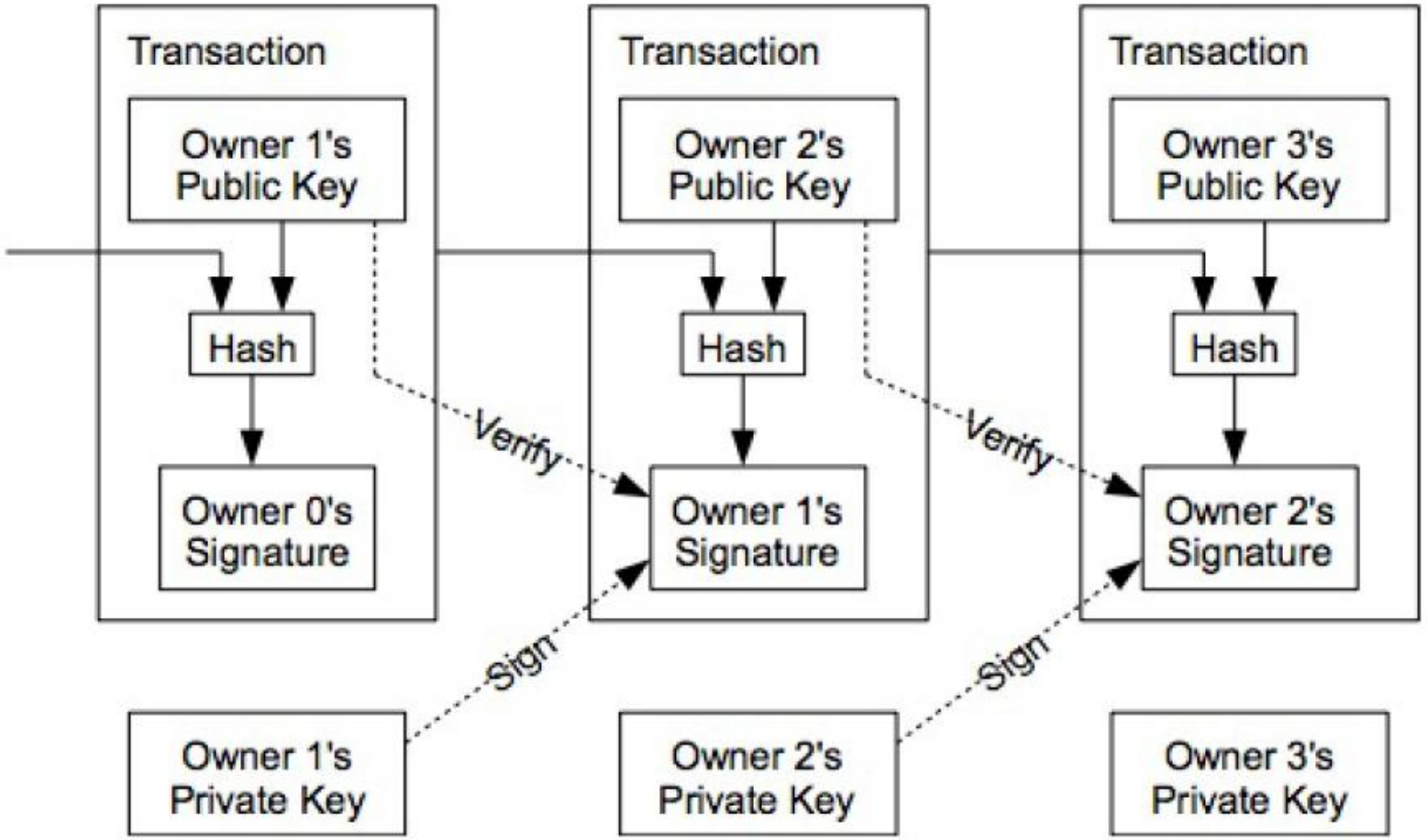
- Cada dirección consta de dos partes matemáticamente correlacionadas:
  - Una dirección pública: Cualquiera que sepa tu dirección pública podrá enviarte bitcoins en cualquier momento.
  - Una clave privada: permite autentificarte, acceder a los fondos que tengas en esa dirección o realizar envíos

Transacción

- Transferencia de un monto de bitcoins de su dueño actual al siguiente.
- Las transacciones de bitcoins se encadenan para poder seguir su historial y verificar validez de fondos.
- Cada transacción contiene:
  - un hash de la transacción anterior
  - Monto a transferir
  - es convalidada con la clave privada del actual dueño.
  - es firmada con la clave pública del próximo dueño.

- No es reversible
- La transacción queda pendiente hasta que es aceptada por la comunidad.

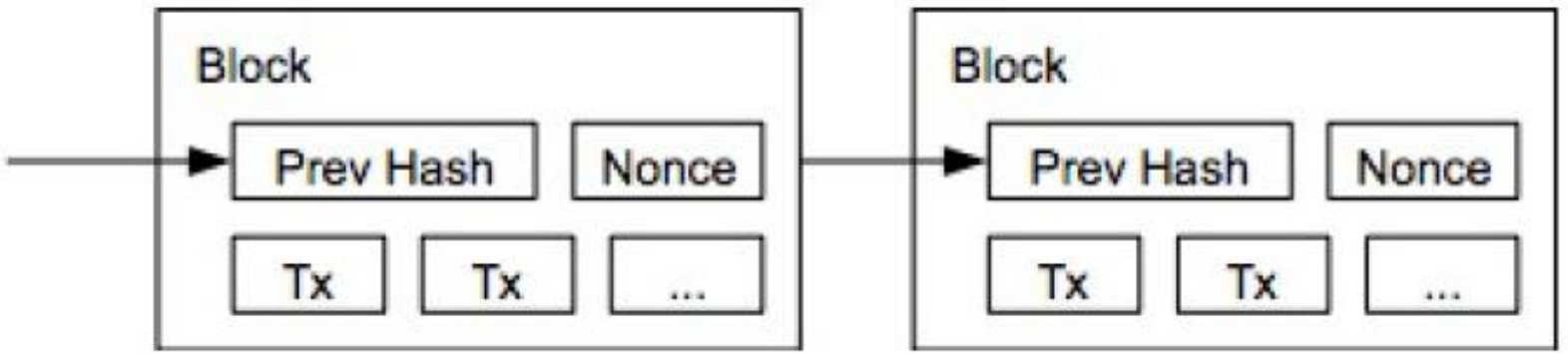
Transacción



Block chain

- Contiene cada transacción procesada, permitiendo verificar la validez de cada transacción.
- Se organizan en bloques que se encadenan secuencialmente mediante un hash.
- Los bloques establecen el historial de transacciones.
- Para poder trabajar en la red se debe contar con la copia completa del block chain
- Contempla mecanismos de compresión.

Block chain



Proof of work

- Cada bloque contiene una prueba de trabajo matemático.
- Son muy difíciles de calcular: miles de millones de cálculos por segundo.
- Los mineros deben hacer estos cálculos antes de que sus bloques sean aceptados por la red y antes de que sean recompensados.
- Cuanta más gente empiece a minar, la dificultad de encontrar bloques válidos se incrementa automáticamente.
- El tiempo promedio de encontrar un bloque es siempre de 10 minutos

Bitcoin – Mining

- Proceso de invertir capacidad de computacional para procesar:
  - Transacciones
  - garantizar la seguridad de la red
  - conseguir que todos los participantes estén sincronizados
- La minería de bloques consiste en calcular el proof of work del mismo
- Buscar un valor para el “nonce” que al hashearse, asegure que el hash comienza con una cierta cantidad de bits en cero.

Generación de bloques en acción

- Nuevas transacciones son transmitidas a todos los nodos.
- Cada nodo recolecta nuevas transacciones en un bloque.
- Cada nodo trabaja en encontrar un proof-of-work difícil para su bloque.
- Cuando un nodo encuentra un proof-of-work, transmite el bloque a todos los nodos.
- Los nodos aceptan el bloque solamente si todas las transacciones en el son válidas y no fueron previamente hechas.
- Los nodos expresan la aceptación del bloque a través de trabajar en

Bitcoin – Recompensas

- El minero compite con el resto de los mineros para crear un bloque valido.
- Al generarlo lo comunica al resto de la red y espera que los demás lo verifiquen y aprueben
- De aprobarse el minero puede obtener 2 beneficios:
  - Bitcoins gratis (monto que con el tiempo se va disminuyendo)
  - Comisiones por transacciones

Perdidas de bitcoins

- Cuando un usuario pierde su dirección privada los bitcoins no pueden ser recuperados
- Aún permanecen en la cadena de bloques al igual que otros bitcoins, pero no pueden transferirse.

Double Spending Problem

- Posibilidad de gastar dos veces una misma moneda digital
- El problema radica en el hecho de al poder duplicar archivos digitales (que representan a la moneda) es fácil duplicar una moneda
- Para sistemas tradicionales existe una tercer parte que verifica que no ocurra (banco, empresa de tarjetas de crédito, etc)

Bitcoin y doble gasto

- Un **usuario común** no puede hacer doble gasto, pues no controla el block chain
- De esa forma no puede validar una transacción.
- Un usuario minero puede intentar introducir transacciones falsas o duplicadas en nuevos bloques para intentar acrecentar su monto de bitcoins
- Un usuario minero puede intentar borrar transacciones pasadas para evitar disminución de sus bitcoins

Doble gasto – controles

- Si se borra una transacción pasada el minero **debe modificar el bloque y recalcular su hash**
- Además debe modificar todos los bloques posteriores para que sus hash sean validos. (bloques encadenados)
- El poder computacional de la comunidad es mayor al individual.

Doble gasto – controles

- Si se introduce o adultera una transacción de una moneda ya gastada en un bloque se generan 2 bloques de cadenas diferentes.
- Solo tendrá validez aquella donde la transacción involucrada es mas antigua.
- Solo se tendrá en cuenta la aceptada por la mayoría.
- La comunidad rechaza la otra