Come funzionano i Bitcoin (e il Dark Web)

Uno dei punti di svolta fondamentali nel corso dell'evoluzione del genere umano è rappresentato dall'invenzione del denaro. Molti animali, infatti, conducono una vita da individui, senza mai costruire una società (pensiamo, per esempio, ai gatti). Altri animali, ma in numero inferiore, hanno imparato a vivere in società più o meno grandi: molte scimmie vivono in vere e proprie famiglie, più o meno organizzate. Ed è naturale che, quando c'è una organizzazione sociale in un gruppo, ogni individuo abbia un compito che serve alla comunità. L'individuo cede qualcosa al gruppo, ed il gruppo ridistribuisce i beni a tutti gli individui. Nasce quindi l'esigenza del baratto: una scimmia raccoglie delle banane, ed un'altra raccoglie dei mango. La prima consegnerà un paio di banane alla seconda in cambio di un frutto di mango, e viceversa. Il baratto, dunque, è una componente fondamentale della società, perché grazie ad esso non è necessario che tutti sappiano fare tutto e ci si può specializzare in un compito particolare sapendo che qualcun altro si occuperà del resto. Gli animali che hanno implementato un modello di società tendono a seguire il meccanismo del baratto. Ma solo un animale è, finora, riuscito a superare il baratto con un trucco: fissare in modo univoco il valore di un certo oggetto. È il concetto di denaro, e l'animale in questione è ovviamente l'uomo. L'acquisto di beni o servizi tramite denaro è, fondamentalmente, una forma di baratto: il cliente prende dal venditore un casco di banane in cambio di un pezzo di metallo che, per convenzione, vale 2 euro. Ma l'uso del denaro ha due vantaggi importantissimi rispetto al baratto: il primo vantaggio è che in questo modo è più facile tenere sotto controllo il valore delle cose. Infatti, un venditore può barattare un caso di banane con un ananas, mentre un altro venditore baratta lo stesso casco di banane con una capra. Ma

difficilmente si potrebbe vendere quel casco di banane per 2 euro in un negozio e 20 euro nell'altro. L'altro grande vantaggio è la possibilità di accumulare il denaro per tempi migliori: non si possono accumulare banane per aspettare di barattarle con qualcos'altro durante l'inverno o addirittura negli anni a venire, perché marcirebbero nel frattempo. Invece, nel caso del denaro, è possibile metterlo da parte per spenderlo in momenti di magra. È il concetto delle pensioni: durante la vita lavorativa si mettono da parte dei soldi, che vengono poi utilizzati per mantenersi quando non si è più in grado di lavorare. E se già così non ci piace molto l'idea di mettere da parte denaro per almeno vent'anni, mettere da parte banane e ananas per due decadi avrebbe risultati decisamente peggiori.

Table of Contents

- I "bug" del denaro
- La soluzione
- Rivest, Shamir, Adleman
- Dai messaggi alle monete
- Ma è davvero sicuro?
- Come si producono
- Vendere online in Bitcoin
- Come si accede al server Onion

I "bug" del denaro

L'introduzione che avete appena letto è, probabilmente, inutile: tutti noi utilizziamo denaro ogni giorno. Ma, proprio perché ne siamo tanto abituati, lo diamo per scontato non pensiamo troppo a come funzioni davvero il meccanismo dell'acquisto, in contrapposizione a quello del baratto.
Ora la domanda è: quali sono le vulnerabilità del denaro? Sicuramente, la contraffazione: fin da quando è apparso per la

prima volta, il denaro è stato soggetto a falsificazioni. I progettisti delle varie monete hanno sempre cercato, nel corso dei secoli, metodi per rendere la vita più difficile possibile ai falsari. Nel corso degli ultimi secoli si sono utilizzate monete realizzate con punzoni particolari, difficili da replicare fedelmente, mentre oggi siamo abituati a vedere sulle banconote filigrane molto particolari. Ma i falsari prima o poi riescono sempre a trovare un metodo per duplicare il denaro.

L'avvento della contabilità digitale ha in buona parte risolto questo problema, ma ne ha introdotto un altro: la tracciabilità, che può generare nella violazione della privacy.

C'è una precisazione da fare: abbiamo parlato di "contabilità digitale" e non di "denaro digitale". Infatti, quando facciamo un acquisto online tramite Visa o Mastercard, il denaro è sempre reale: la contabilità delle transazioni è digitale. In pratica, non cambia il meccanismo di pagamento: alla fin fine se acquistiamo una penna su ebay, il denaro che paghiamo arriverà in mano al venditore. Possiamo schematizzare in questo modo: noi versiamo dei contanti alla nostra banca, la somma entra nel nostro conto corrente online, paghiamo la cifra dovuta al commerciante, tale cifra viene spostata sul conto del venditore, e questo va alla sede della sua banca per prelevare, tramite bancomat, il denaro. I denaro rimane sempre sotto forma di monete o banconote, anche se le banconote che il venditore si ritroverà in mano non saranno fisicamente le stesse che noi avevamo versato in banca. Il denaro in circolo è sempre rappresentato da banconote fisiche.

Quello che cambia è il modo in cui le transazioni vengono registrate: con la contabilità "classica" è facile perdere le tracce dopo un paio di intermediari. Per esempio, se acquistiamo un ombrello pagando in banconote innanzitutto non è possibile sapere che siamo stati davvero noi ad eseguire l'acquisto e non, piuttosto, qualcun altro. In secondo luogo, il commerciante avrà segnato la cifra che gli abbiamo corrisposto tra le sue entrate, ma a sua volta è difficile

capire (anche se non impossibile) quale parte del denaro che gli abbiamo corrisposto rappresenta davvero il valore dell'ombrello e quale parte, invece, è il guadagno personale del commerciante che nulla ha a che vedere con il valore dell'oggetto in se.
Riassumendo, il problema sta nel fatto che gli acquisti sono facilmente tracciabili, e questo può rivelarsi un problema soprattutto in paesi privi delle libertà fondamentali. Un dissidente di un regime autoritario ha sicuramente il conto corrente sotto controllo.
Inoltre, il sistema è centralizzato, e i gestori possono commettere violazioni sui correntisti: l'esempio più noto è quello di Julian Assange, che dopo avere messo in imbarazzo il governo degli Stati Uniti si è ritrovato con i conti correnti congelati, impossibilitato ad eseguire qualsiasi pagamento, senza che vi fosse un provvedimento del tribunale ma per spontanea decisione degli istituti bancari.

La soluzione

Dunque, il denaro "analogico" ha due punti deboli: la falsificazione e la tracciabilità, e l'entità dei problemi varia a seconda del fatto che si usi una contabilità "tradizionale" oppure "digitale". Qualcuno, però, ha pensato ad un modo per risolvere queste vulnerabilità: passare completamente al digitale. Il programmatore Satoshi Nakamoto (è uno pseudonimo, non si conosce il nome reale di questa persona) ha proposto, nel 2008, una idea di **cryptocurrency**, cioè di "moneta virtuale cifrata". Il concetto stesso di cryptocurrency era già stato suggerito nel 1998, ma nessuno aveva mai realizzato prima una implementazione ben funzionante come quella di Satoshi Nakamoto, conosciuta col nome di Bitcoin.

I bitcoin sono legali?
Ci si potrebbe chiedere se i Bitcoin siano legali o meno. In effetti, essendo anonimi e non rintracciabili vengono utilizzati soprattutto per acquisto di materiale illecito o per ripulire denaro sporco. In realtà, però, si tratta di un bene di consumo come potrebbe esserlo una fotografia digitale. Quindi non è facile per le autorità proibirne l'uso. In Europa e negli Stati Uniti, quindi, l'uso dei Bitcoin come forma di pagamento è tollerato, anche se i governi lo sconsigliano visto che è da considerarsi sempre un investimento ad alto rischio. È capitato che le autorità statunitensi abbiano chiuso alcuni siti che consentivano di tenere portafogli Bitcoin online, requisendo le monete digitali: l'operazione è però avvenuta soltanto nei casi in cui era stata dimostrata l'esecuzione di attività illegali tramite i siti web in questione.

Come funziona una moneta digitale? Abbiamo visto che una moneta è in realtà un qualsiasi oggetto a cui viene dato un valore preciso, e che viene scambiata con altri oggetti come in una sorta di baratto. Quindi, per realizzare una moneta virtuale potremmo utilizzare un qualsiasi oggetto digitale, ed assegnargli un valore: una immagine, un file audio, una stringa di testo. Naturalmente, questi esempi non vanno bene, perché sono troppo facili da replicare: è necessario qualcosa di univoco, cioè una tipologia di oggetto in cui ogni esemplare è riconoscibile e non duplicabile (un meccanismo, quindi, simile a quello dei numeri di serie sulle banconote). La matematica ci viene in aiuto, ed ecco quindi il concetto di cryptocurrency. L'idea è di utilizzare particolari funzioni matematiche che consentono di calcolare coppie di numeri tra essi collegati ma tali da non poter risalire ad uno dei due anche se si conosce l'altro, di modo che l'unico a conoscerli entrambe sia chi li ha calcolati. È la stessa logica della

crittografia asimmetrica.

Rivest, Shamir, Adleman

Tutti noi abbiamo utilizzato la crittografia simmetrica: si sfrutta la stessa chiave per cifrare e per decifrare un messaggio. È il caso di una cifratura del tipo Cesare (cioè le lettere del testo vengono spostate, per esempio la A diventa B, la B diventa C e così via). Si tratta della forma più semplice e comune di cifratura, molti algoritmi di crittografia di uso comune sono “semplici” (per esempio quello degli archivi Zip o Rar). Questo tipo di cifratura ha un grosso svantaggio: visto che la chiave necessaria per decifrare un messaggio è la stessa usata per criptarlo, è fondamentale che il mittente invii al destinatario anche la chiave, oltre al testo cifrato. E questo complica le cose, perché se era davvero necessario proteggere il messaggio con la crittografia, non si può certo spedire la chiave senza alcuna misura di sicurezza: se qualcuno intercetta la posta, potrebbe ottenere sia il testo che la chiave, risalendo quindi al contenuto originale in chiaro (cioè non cifrato). Una soluzione al problema della distribuzione della chiave è dato dalla crittografia asimmetrica, nella quale la chiave usata per cifrare e quella necessaria a decifrare il messaggio sono diverse. Tale meccanismo era stato ipotizzato dai crittografi Diffie, Hellman, e Merkle, ed è stato reso possibile dall’algoritmo sviluppato dai matematici Rivest, Shamir, Adleman (che è per l’appunto chiamato algoritmo RSA). Come è possibile avere due chiavi differenti per cifrare e decifrare un messaggio? In realtà è piuttosto semplice, lavorando con l’aritmetica dei moduli: per capirlo meglio, vediamo come funziona l’algoritmo RSA.
Per poter utilizzare questa cifratura abbiamo innanzitutto bisogno di due numeri primi, che chiameremo **p** e **q**. In teoria, questi dovrebbero essere molto grandi (altrimenti sarebbe troppo facile riuscire a scoprirli). Ci serve, poi un altro

numero primo, più piccolo degli altri due: lo chiameremo **e**. Adesso possiamo calcolare il numero **N** come un semplice prodotto tra **p** e **q**. La chiave pubblica sarà data semplicemente dalla coppia **N** ed **e**. Quella privata, invece, si basa numeri **N** e **d**. Qui c'è un piccolo problema: il numero **d** si deve calcolare in modo tale che

*(e*d) mod ((p-1)*(q-1)) = 1*

possiamo, per semplificare, chiamare

(p-1)(q-1) = phi*

La relazione da soddisfare per ottenere **d** deve quindi essere:

*(e*d) mod (phi) = 1*

La cosa migliore per trovare **d** è andare per tentativi finchè non si identifica un numero in grado di soddisfare l'equazione. L'operazione "mod" è quella di modulo (cioè il resto della divisione). Per esempio,

7 mod 5 = 2

Poichè 7 diviso 5 fa 1 con il resto di 2. Ovviamente, **d** non dovrà mai essere rivelato perchè serve a decifrare i messaggi cifrati con e.

Come si può capire, il punto debole della crittografia RSA sta nel fatto di basare la propria sicurezza sulla difficoltà di scomporre il numero **N** in fattori primi e quindi risalire a **p** e **q**. Il problema è che la potenza dei computer aumenta ogni anno, e con essa la loro velocità nel fattorizzare un numero. Per tale motivo si cerca continuamente di costruire chiavi sempre più grandi: è una sorta di grande corsa contro il tempo in cui, per ora, i "buoni" sono avvantaggiati rispetto ai "cattivi". Ma è probabile che entro qualche decina di anni la tendenza sarà invertita, e ci resterà solo la crittografia a blocco monouso come strumento per proteggere la privacy. Ad ogni modo, per cifrare usiamo (per ogni lettera) questa relazione:

C = T^e mod N

e per decifrare quest'altra:

T = C^d mod N

Dove **C** è il testo cifrato e **T** quello in chiaro.

L’RSA è stata scoperta due volte

Rivest, Shamir, ed Adleman non sono stati i primi a scoprire la crittografia asimmetrica. Ma sono stati i primi a poterlo rendere noto. La crittografia tipo RSA venne infatti ideata da James Ellis, Clifford Cocks, e Malcolm Williamson nel 1974-1975. questi erano, però, dipendenti del centro segreto di crittografia GCHQ (servizi segreti del Regno Unito). Nonostante la loro scoperta non fosse stata immediatamente utilizzata dall’intelligence britannica, anche perchè i calcolatori non erano ancora in grado di eseguire conti così complessi, essi furono obbligati a mantenere il silenzio sulla crittografia asimmetrica, ed assistere quindi impotenti alla riscoperta dell’algoritmo da parte dei due trii di crittografi e matematici che abbiamo citato nell’articolo. Ovviamente, ciò non toglie merito ai ricercatori che oggi ricordiamo come scopritori “ufficiali” di questa tecnologia, perchè essi hanno eseguito le loro ricerche in totale autonomia, partendo da zero. Dovremmo, però, ricordare anche i dipendenti del GCHQ, che non hanno mai potuto ricevere i dovuti onori.

L’RSA è la più importante forma di cifratura attualmente esistente. Fondamentalmente tutta la sicurezza informatica si basa su questo algoritmo (per esempio la cifratura SSL delle pagine web, la firma digitale, ecc…). In particolare, è importante capire come funziona il meccanismo di firma digitale. Nel caso della crittografia, si rende pubblica la chiave per cifrare, mentre rimane privata la chiave per decifrare. In questo modo tutti possono scrivere un messaggio criptato ad una persona, ma solo questa persona potrà leggerlo (nemmeno il mittente originale potrà più decifrare il messaggio perché dispone solo della chiave pubblica). Invece, nel caso della firma digitale, si fa esattamente l’opposto. In pratica, la chiave utilizzata per cifrare è privata, l’altra è pubblica. Questo fa sì che un utente possa prendere un documento e cifrarlo con la propria chiave privata: l’operazione garantisce che a scrivere il documento sia stato

proprio questo utente, perché è l'unico a conoscere la chiave di cifratura. Chiunque, invece, può leggere il documento utilizzando la chiave pubblica per decifrarlo: il meccanismo garantisce anche che nessuno abbia modificato il testo dopo l'apposizione della firma perchè, se il testo viene modificato, l'algoritmo di decifrazione restituirà un testo errato.

Dai messaggi alle monete

Ora che abbiamo capito come funziona la crittografia asimmetrica per i messaggi, sarà facile comprendere il meccanismo su cui si basa Bitcoin. La rete Bitcoin è costruita come una rete P2P (come Kadmilla, per esempio). Ogni utente della rete ha diverse (potenzialmente illimitate) coppie di chiavi, che può raccogliere in un portafoglio digitale. In ciascuna di queste coppie, la chiave pubblica è resa nota a tutti e funziona da punto di invio o di ricezione per i vari pagamenti: è in un certo senso un po' come un IBAN. Per tale motivo la chiave pubblica, che per essere precisi è costituita da 33 caratteri (scelti tra lettere e numeri), viene chiamata indirizzo bitcoin. Invece, la chiave privata corrispondente è nota esclusivamente al suo proprietario e serve a consentire il pagamento (cioè la cessione, non la riscossione). In questo modo, solo il proprietario della coppia di chiavi può davvero autorizzare la cessione di alcune sue monete Bitcoin. Un Bitcoin viene "creato" come ricompensa per il calcolo di un blocco (vedremo tra poco che significa) ed è una stringa di testo, contenente l'indirizzo del proprietario, cifrata con la chiave privata del proprietario stesso.
In seguito, se il proprietario vorrà cedere la sua moneta a qualcun altro per eseguire un acquisto, dovrà aggiungere ad essa l'indirizzo (la chiave pubblica) del nuovo proprietario e firmare il tutto con la propria chiave privata. Il fatto che si debba cifrare con la propria chiave privata fa sì che solo l'attuale proprietario possa porre la firma necessaria a concludere l'acquisto. Ciò significa che è sempre possibile

risalire alla storia di una moneta: tramite l'ultima chiave pubblica è possibile decifrare la parte criptata. Questa è costituita da due parti: un'altra chiave pubblica e un altro "blob" criptato (da decifrare con l'indirizzo bitcoin appena citato). A sua volta, questo blob contiene un ulteriore indirizzo bitcoin e relativo blocco di testo criptato con esso. È quindi possibile proseguire fino alla stringa di testo originale, ricostruendo l'intera storia della moneta dal suo ultimo acquirente fino al creatore ottenendo tutti i vari indirizzi bitcoin attraverso cui la moneta in questione è passata.

In pratica, ogni moneta Bitcoin esiste in quanto oggetto di una transazione economica: è addirittura costruita dall'insieme dei vari indirizzi che l'hanno posseduta. In fondo, ciò che conta di una moneta è poter eseguire delle transizioni: sono proprio queste il punto focale del meccanismo, quindi incentrare su queste il protocollo Bitcoin è la soluzione più semplice. Le transizioni sono pubbliche, anzi: quando si esegue un pagamento, le informazioni sulle monete (quantità di Bitcoin spesi, indirizzo del mittente e

del destinatario) vengono inviate all’intera rete. Inoltre ogni client Bitcoin, per poter funzionare, deve scaricare dalla rete gli aggiornamenti in tempo reale sulle transizioni eseguite nel mondo. In questo modo è impossibile che un utente spenda due volte una propria moneta: quando esegue la prima transazione, tutti sanno che quel particolare Bitcoin è passato ad un altro utente e dunque non è più di sua proprietà.

Ma è davvero sicuro?

Si potrebbe obiettare che se davvero è possibile conoscere tutti gli indirizzi proprietari di una moneta, e tutti conoscono tutte le monete esistenti (grazie al database di informazioni), il sistema sia tracciabile. In realtà, però, gli indirizzi bitcoin vengono generati in modo casuale, e non possono essere collegati direttamente ad una persona: è infatti consigliabile utilizzare un indirizzo diverso per ogni transazione. Facciamo un esempio: se vogliamo permettere ai lettori del nostro blog di inviarci un pagamento tramite bitcoin, dovremo fornire pubblicamente un indirizzo a cui far arrivare i pagamenti. Ciò significa che tutti, Guardia di Finanza e Polizia Postale incluse saranno in grado di conoscere questo indirizzo ed i suoi movimenti, compresi e soprattutto quelli futuri: potrebbero facilmente scoprire quali oggetti servizi acquisteremo con quelle monete. La soluzione consiste nell’avere a disposizione diversi indirizzi bitcoin ma non dirlo a nessuno. Dal momento che le chiavi pubbliche non si possono collegare alle persone in modo automatico, ma solamente se il proprietario decidere di farlo sapere a tutti, è sufficiente avere un indirizzo “pubblico”, a cui tutti possono inviare denaro, e diversi indirizzi “privati”su cui trasferire le monete che arrivano a quello pubblico. In questo modo, eventuali inquirenti potrebbero sapere quante monete ci sono state cedute, ed anche quante ne abbiamo inviate ad altri indirizzi Bitcoin. Ma non potrebbero

in alcun modo sapere chi sia il reale proprietario di questi indirizzi di arrivo: certo, potrebbero sospettare che si tratti di indirizzi che appartengono comunque a noi e che sia tutta una messa in scena per far perdere le tracce, ma non potrebbero dimostrarlo. Va anche detto che in realtà la "cronologia" di una moneta si può ricostruire solo grazie al database delle transazione, e non direttamente dalla moneta stessa. Questo perchè, di norma, invece di cifrare l'intero testo che costituisce la moneta per eseguire la firma che garantisce una transazione economica, si cifra soltanto un hash (con algoritmo SHA-2) del testo in questione. Un hash è un'altra stringa di testo ottenuta con una funzione di hash dalla stringa che rappresenta la moneta. La funzione di hash è per definizione non iniettiva, quindi non è biettiva e non si può invertire. Di conseguenza, dalla moneta si ottiene l'hash, ma dall'hash non si può ottenere la moneta. Tuttavia, ogni moneta realizza un unico hash, e questo può essere realizzato da una solo moneta: due stringhe di testo, cioè due monete bitcoin, differenti non possono generare lo stesso hash. Ciò significa che chi possiede una moneta può risalire fino all'hash generato dal suo precedente proprietario, ma lì deve fermarsi: non può ricostruire il contenuto della moneta prima che questa arrivasse nelle mani della persona da cui gli stesso l'ha ottenuta. Certo, può comunque risalire all'intera cronologia senza nemmeno bisogno di avere la moneta "in mano": gli basta controllare il database comune delle varie transazioni.

Come si producono

L'ultima cosa che dobbiamo ancora scoprire sui Bitcoin è in che modo si possono ottenere. In fondo, per ogni moneta è fondamentale stabilire in che modo possa essere prodotta, per indicare anche un limite di produzione (anche nella realtà, la banca centrale europea non può stampare tutti gli euro che vuole, ci sono delle regole da rispettare).

Di solito nessuno si preoccupa di come vengano creati i bitcoin, perché esistono già strumenti in grado di farlo. Il fatto è che per degli sviluppatori è fondamentale saperlo, soprattutto perché man mano che passa il tempo servono sistemi sempre più efficienti per produrre nuovi blocchi, e quindi è necessario che qualcuno sappia come programmare questi "generatori di blocchi".
Cos'è un blocco? Abbiamo detto che tutte le transizioni vengono comunicate, sottoforma di messaggio, all'intera rete Bitcoin. E la rete memorizza questi messaggi in diversi "blocchi", che sono quindi semplicemente dei contenitori realizzati periodicamente.
Questo permette a nuovi client di scaricare facilmente l'intero elenco delle transizioni mai eseguite: basta ottenere tutti i blocchi, che tra l'altro sono identificabili (ci sono i blocchi della settimana scorsa, quelli di due mesi fa, quelli di tre anni cinque mesi e tre giorni fa, eccetera).
I Bitcoin vengono forniti agli utenti come ricompensa per avere risolto un problema matematico, operazione chiamata "mining" (letteralmente "scavare in miniera"). Il problema in questione consiste nell'identificare un numero (chiamato **nonce**) tale che dopo essere stato sottoposto due volte all'algoritmo di hash SHA-2 si ottenga una stringa che inizia con un certo numero di zeri. Naturalmente, visto che la funzione hash non è invertibile, si deve procedere per tentativi (un po' come nel caso del brute force). Il numero di zeri che devono essere presenti all'inizio della stringa viene variato automaticamente per rendere l'operazione più semplice o più complessa in modo che venga sempre generato, in media, un nuovo blocco ogni 10 minuti. La difficoltà di trovare il numero **nonce** aumenta esponenzialmente con la quantità di zeri che devono essere presenti all'inizio della stringa ottenuta con il doppio hash.
Ogni nodo della rete, rappresentato da un utente con il proprio client, si occupa quindi di calcolare questo numero ed utilizzarlo per ottenere assieme al contenuto del blocco su cui sta lavorando un hash. Quando riesce ad ottenere l'hash

che inizia con il giusto numero di zeri, comunica l'avvenuta scoperta a tutta le rete. Quando gli altri nodi della rete riconoscono che la soluzione proposta è corretta, "archiviano" il blocco, che entra dunque a far parte del passato, e cominciano ad inserire le prossime transizioni che riceveranno in un blocco completamente nuovo. L'insieme in ordine cronologico dei vari blocchi è chiamato "catena dei blocchi". In pratica, la rete Bitcoin si comporta in questo modo:

- Quando due utenti eseguono una transazione, questa viene comunicata a tutti i nodi
- Ogni nodo minatore raccoglie le nuove transizioni di cui viene informato in un blocco
- Ogni minatore calcola la funzione per risolvere il problema matematico (questo è il mining vero e proprio)
- Quando un nodo trova la soluzione, cioè il numero **nonce**, la invia a tutto il resto della rete
- I nodi accettano il blocco soltanto se le transizioni in esso contenute sono valide
- Il minatore ottiene una ricompensa in bitcoin per il blocco contenente la soluzione che ha appena scoperto
- I nodi dichiarano di avere accettato il blocco ricevuto e cominciano a lavorare su un altro blocco utilizzando l'hash del blocco appena accettato.
- Si ricomincia da capo

Per ogni blocco vengono forniti dei Bitcoin, in numero decrescente nel tempo. Infatti, nel 2008 per ogni blocco venivano forniti 50 Bitcoin, mentre dal 2012 ne vengono prodotti 25 (e così sarà fino al 2016). La ricompensa verrà dimezzata ogni 4 anni fino ad arrivare a zero: secondo questo meccanismo, sarà possibile generare al massimo 21 milioni di monete. Visto che la complessità di calcolo aumenta sempre più, è oggi praticamente impossibile ottenere qualcosa con il proprio computer: è necessario utilizzare dei minicomputer dedicati (per esempio **ASIC**) oppure collaborare con altri

minatori (per esempio tramite il sito **www.bitminter.com**). Nel caso della collaborazione, il sito si occupa di distribuire i bitcoin tra i vari utenti che hanno contribuito a risolvere il problema sul blocco. Ovviamente, visto che vengono forniti solo 25 Bitcoin e probabilmente ci sono ben più di 25 collaboratori tra cui distribuirli, ogni utente otterrà una frazione di moneta. In pratica, invece di ottenere 1 Bitcoin, riceverà 0,00X Bitcoin. Questo non è un problema, sia perché i Bitcoin possono essere divisi fino all'ottava cifra decimale, sia perché il valore dei Bitcoin può aumentare man mano che la gente li richiede.

Il valore dei Bitcoin

Il punto dolente dei Bitcoin sta proprio nel valore della moneta, ed è legato alla loro logica decentralizzata. Il motivo per cui i Bitcoin sono da considerarsi un investimento ad alto rischio è l'incredibile fluttazione che il suo valore ha avuto e continua ad avere. Un bitcoin può passare dal valore di 50 euro a quello di diverse centinaia di euro nel giro di pochi mesi. Un andamento semplicemente impossibile per le monete tradizionali. Il fatto è che, non esistendo un organismo centrale, non esiste nessuno che posso stabilire uno standard per il valore della moneta e che possa quindi prendere le dovute misure contro l'inflazione e la deflazione. Il valore dei Bitcoin è legato quasi esclusivamente al rapporto di domanda ed offerta: sono dunque gli utenti, con le loro decisioni di comprare o vendere le proprie monete che fanno salire o scendere il prezzo di un Bitcoin. E, considerando che ogni utente ragiona con la propria testa e per motivi personali, la variazione del valore della moneta è praticamente casuale.

Quando i bitcoin non potranno più essere prodotti (perchè la ricompensa per ogni blocco sarà diventata pari a zero) i nodi

potranno comunque continuare a realizzare i blocchi finanziandosi in altri modi, per esempio guadagnando dalle varie transazioni gestite.

Vendere online in Bitcoin

Per accettare pagamenti in Bitcoin sul proprio sito, è possibile sfruttare diversi servizi, come accept-bitcoin, che di fatto sono il PayPal per Bitcoin. Basta aggiungere al proprio sito un codice di questo tipo:

Nell'esempio stiamo vendendo 1Kg di Nduja per l'equivalente in Bitcoin di 20 euro. C'è anche una procedura guidata per realizzare questo codice, la si trova sul sito https://accept-bitcoin.net/bitcoin-payment-button.php. Ovviamente, pubblicare sul proprio sito web del materiale da pagare con Bitcoin ha poco senso: si viene facilmente tracciati. Molto meglio realizzare il proprio sito nel Dark Web, con Tor. Il bello di avere un server web che funziona sulla rete Tor, cioè un cosiddetto sito web Onion, è che non siamo identificabili: gli utenti del nostro sito web non possono risalire al nostro indirizzo IP reale, e quindi non ci possono identificare. Allo stesso modo, nessuno (noi compresi) può identificare gli utenti del nostro sito: tutti sono assolutamente anonimi.

Quando si scarica Tor Browser, la prima finestra che appare permette di configurare la connessione ad internet: nel caso ci si trovi dietro ad un proxy (come nelle reti aziendali), può essere necessario cliccare su Configure. Altrimenti, basta cliccare su Connect per iniziare la connessione a Tor. Il servizio Tor rimarrà attivo finché Tor Browser è aperto: se lo si chiude, anche la connessione a Tor viene terminata.

I siti web con dominio .onion sono per l'appunto anonimi e costituiscono il dark web, che è un sottoelemento del deep web. Il deep web più in generale rappresenta tutti quei siti non rintracciabili dai normali motori di ricerca, ma all'interno del deep web il dark web è statisticamente la parte più ampia e interessante (esistono comunque anche siti privati, che non sono accessibili da chiunque pur non essendo anonimi) e per questo i due termini sono spesso usati come sinonimi. Questo anonimato offerto dai server .onion può non essere particolarmente importante per un normale cittadino italiano, ma è fondamentale per chi si trova in paesi che limitano la libertà di stampa: un cittadino ucraino poteva (durante la rivoluzione) pubblicare un sito Tor con notizie sui crimini del governo, senza il rischio di essere scoperto ed arrestato. Per fare un altro esempio, la procedura di invio

dei file da parte degli informatori a Wikileaks funzionava proprio usando un server web Tor.

Per abilitare il proprio Hidden Service è necessario aprire il file che si trova nella cartella **Browser/TorBrowser/Data/Tor/torrc** con un editor di testo, e inserire alla fine di esso le righe https://pastebin.com/EEa4PegH. In questo modo si indica come server quello che viene ospitato sulla porta 80, e come cartella dei dati /home/luca/tor-browser_en-US/hidden_service/. Ovviamente, si può scegliere una cartella qualsiasi, e una qualsiasi porta (FTP funzionerebbe sulla porta 21).

Considerata la struttura della rete Tor stessa il client Tor può essere utilizzato sia per accedere a dei servizi che per proporre i propri servizi al resto della rete. Infatti Tor è un protocollo di Onion routing, un tipo di rete simile a una VPN, con una differenza importante: ci sono molti passaggi intermedi. Una VPN è di fatto una sorta di rete locale molto estesa sul territorio, tutti i computer si collegano a uno stesso server centrale che funziona come un router domestico e

tiene assieme tutti i vari computer permettendo loro di condividere informazioni e un unico indirizzo IP pubblico. L'indirizzo IP pubblico è quello del server, quindi ogni client appare sul web con il suo indirizzo. Nell'Onion routing esistono molti passaggi intermedi: in pratica, un client si collega a un server, il quale si collega a un altro, poi un altro, e così via fino a un server finale che funziona come "punto di uscita" (un exit node). Il client apparirà sul web con l'indirizzo IP del punto di uscita, ma nessuno dei vari punti intermedi può risalire a chi sia il client e quale il nodo di uscita: ogni nodo intermedio della rete sa soltanto chi c'è prima di lui e chi dopo, ma non sa in che punto esatto della rete si trovi, quindi non sa se sta dialogando con il primo o l'ultimo computer della sequenza.

Ora è ovviamente necessario installare il server: se si vuole un server web, l'opzione probabilmente più scontata è Apache2, che può essere installato su sistemi Debian con il comando: *sudo apt-get install apache2 php5 libapache2-mod-php5 php5-mcrypt* Dando il comando *cat /etc/apache2/sites-enabled/000-default.conf* la riga DocumentRoot indica il percorso nel quale si possono inserire i vari file, per esempio la cartella

/var/www/html/. Il file index.html dovrebbe già esistere.

L’idea ricorda un po’ le reti mesh, con la differenza che nelle reti mesh solitamente è possibile per tutti i computer essere punti di uscita per qualcun altro e non si può scegliere il proprio punto di uscita. Invece, nell’Onion routing i punti di uscita sono solo alcuni dei computer coinvolti, e un client può cambiare exit node se desidera apparire con un indirizzo IP differente da quello che ha avuto finora.
Inoltre, nell’Onion routing c’è un apposito sistema crittografico: un sistema a cipolla (il nome non è casuale, del resto). Ogni router intermedio si presenta al client con una chiave crittografica pubblica, che può essere usata per crittografare i pacchetti. Immaginiamo che un client riceva le chiavi di 4 diversi router: il client preparerà un pacchetto crittografandolo con la chiave 4, la chiave 3, la 2, e infine la numero 1. Il pacchetto verrà poi inviato ai vari router in sequenza, che lo potranno decifrare man mano: il primo router toglierà soltanto la crittografia con la propria chiave privata 1, il secondo con la chiave privata 2, eccetera. In questo modo, gli unici a poter vedere i dati non crittografati sono il client e l’exit node (e se si utilizza un protocollo come HTTPS nemmeno l’exit code può vedere i dati completamente decifrati, potrà farlo soltanto il vero destinatario). Le varie crittografie sono infatti applicate a strati come in una cipolla, e devono essere tolte esattamente in quell’ordine. Se qualcuno provasse a intromettersi il procedimento salterebbe e i dati diventerebbero automaticamente illeggibili, rendendo quindi tutta la comunicazione privata e di fatto anonima perché nessuno sa chi sia davvero ad avere cominciato la connessione.

Come si accede al server Onion

Per accedere ad un server Onion non si utilizza un indirizzo IP, ma un nome di dominio che viene creato in modo casuale sulla base di una altrettanto casuale e univoca chiave crittografica. Si tratta della chiave crittografica pubblica che permette ad altri client di crittografare, come abbiamo spiegato, i pacchetti di dati in modo che soltanto il nostro server sia in grado di leggerli usando la chiave privata di decifratura. Tutto quello che si deve fare per rendere il proprio client accessibile dalla rete Tor è selezionare le porte da “aprire” al resto del dark web (nel tutorial suggeriamo come fare per la porta 80).

Per conoscere l’indirizzo del sito web sulla rete Onion, accessibile anonimamente tramite rete Tor, basta leggere il contenuto del file **tor-browser_en-US/hidden_service/hostname**. È l’indirizzo che si può condividere, anche sui motori di ricerca. Inserendo l’indirizzo su un Tor Browser è possibile visualizzare il sito appena creato. Per assicurarsi che il

sito sia accessibile solo tramite rete Tor, e non sul web, è una buona idea chiudere la porta 80 sul proprio router.

C’è però una differenza importante con i server che si abilitano sul proprio computer per l’accesso dal web convenzionale: non si passa attraverso il meccanismo del NAT, perché il client Tor è già connesso direttamente alla rete Onion. Le normali reti locali casalinghe sono infatti gestite da un router, e sono quindi “nattate”: questo significa, in parole povere, che l’unico computer visibile direttamente da internet è il router stesso. Gli altri dispositivi della rete non sono normalmente visibili, e le loro porte di comunicazione (il server web utilizza la numero 80) sono chiuse. È quindi fondamentale, se vogliamo che il nostro server sia raggiungibile da internet, aprire la porta 80 del nostro computer. Questa procedura è detta “port forwarding”: se controlliamo il manuale del nostro router troveremo sicuramente una pagina in cui viene spiegato come eseguirla. Tuttavia, su rete Tor il server funziona anche se non abbiamo abilitato il port forwarding, perchè la connessione viene gestita direttamente dalla rete Onion. Anzi: è molto meglio non aprire la porta 80 sul router. Così, l’unico modo per accedere al server sarà tramite la rete Tor: un utente (e questo vale anche per i programmi di scansione automatica come Echelon o Prism) che si trova su internet (e che quindi può leggere il nostro vero indirizzo ip) non sarebbe in grado di entrare nel sito. Se aprissimo la porta sul nostro router casalingo, il nostro server diventerebbe accessibile dal web normale, non anonimo, e prima o poi qualcuno lo troverebbe (potrebbe finire sul motore di ricerca Shodan). Se la teniamo chiusa, l’unico modo per arrivare al nostro server web sarebbe proprio attraverso il client Tor che abbiamo attivo sul computer, quindi saremmo protetti dal meccanismo di anonimato.

DE GRUYTER OPEN Proceedings on Privacy Enhancing Technologies ; 2016 (4):454–469

Mauro Conti, Stephen Crane, Tommaso Frassetto, Andrei Homescu, Georg Koppen, Per Larsen, Christopher Liebchen, Mike Perry, and Ahmad-Reza Sadeghi

Selfrando: Securing the Tor Browser against De-anonymization Exploits

Abstract: Tor is a well-known anonymous communication system used by millions of users, including journalists and civil rights activists all over the world. The Tor Browser gives non-technical users an easy way to access the Tor Network. However, many government organizations are actively trying to compromise Tor not only in regions with repressive regimes but also in the free world, as the recent FBI incidents clearly demonstrate. Exploiting software vulnerabilities in general, and browser vulnerabilities in particular, constitutes a clear and present threat to the Tor software. The Tor Browser shares a large part of its attack surface with the Firefox browser. Therefore, Firefox vulnerabilities (even patched ones) are highly valuable to attackers trying to monitor users of the Tor Browser.

In this paper, we present selfrando—an enhanced and practical load-time randomization technique for the Tor Browser that defends against exploits, such as the one FBI allegedly used against Tor users. Our solution significantly improves security over standard address space

1 Introduction

The Tor Project provides a suite of free software and a worldwide network designed to facilitate anonymous information exchange and to prevent surveillance and fingerprinting of these interactions. The Tor network is open to anyone and widely used by civil rights activists, whistleblowers, journalists, citizens of oppressive regimes, etc. Many sensitive websites, including the late Silk Road black market, are only accessible over Tor. Consequently, the Tor Network is continually facing de-anonymization attacks by law enforcement, intelligence agencies, and foreign nation states. A de-anonymization attack aims to disclose information, such as the identity or the location, of an anonymous user. While many de-anonymization attacks rely on weaknesses in the network protocol, they often require that adversaries control a large number of Tor nodes [26] or only work in a lab environment [39].

Le ultime versioni di Tor Browser contengono Selfrando, sistema sviluppato dall’università di Padova per proteggere il programma dall’esecuzione remota di codice