

Blockchain

Rapport

|  |  |
| --- | --- |
| **Team n°** | 7 |
| **Rédacteurs** | Yoann BILIATO, Antonin BROQUET, Lucie MERCENNE, Michaël PERRIN |
| **Date de création** | 25 novembre 2017 |
| **Statut** | Intermédiaire, pour revue. |
| **Version** | 1.0 |

# 

# SOMMAIRE

[**SOMMAIRE**](#_suhfxzcw2kgy) **1**

[**ACRONYMES ET DÉFINITIONS**](#_qab1hmu2fdsu) **2**

[**RÉSUMÉ**](#_xwcw6jq3no22) **3**

[**1 - INTRODUCTION**](#_ynvrhec3gj8l) **4**

[1.1 - Contexte](#_z686mw1jw20e) 4

[1.2 - Objectifs](#_kqx2oz24j0op) 4

[**2 - BLOCKCHAIN**](#_vt2uz6buo5oq) **5**

[2.1 - QU’EST-CE QU’UNE BLOCKCHAIN?](#_3e8mj475xm49) 5

[2.1.1 Hash](#_z6kdks74eko2) 5

[2.1.2 Chiffrement asymétrique](#_5hjo3i6dukpy) 5

[2.1.3 Transactions](#_ypbwz3wxb4fv) 6

[2.1.4 Bloc](#_z905xebrz09k) 6

[2.1.5 Minage](#_t5bj21evnji3) 7

[2.2 - ETHEREUM](#_djsu7ln3vd9) 7

[2.2.1 - L’EVM](#_k6tfryz14uam) 7

[2.2.2 - Smart contract](#_wuyvd1lalvor) 8

[**3 - DEMONSTRATEUR**](#_j4tt1jymynn3) **9**

[3.1 - INTRODUCTION](#_1cn63ccm67a4) 9

[3.2 - ARCHITECTURE](#_4xnxmz1o2n59) 9

[3.3 - APPLICATION](#_d7xjonijxr0r) 11

[**4. ANALYSE ET CONCLUSION**](#_srvsbj4yi6bk) **13**

[**RÉFÉRENCES**](#_xtkifb2qyj4k) **14**

[**Annexe A - Illustration transaction Bitcoin**](#_bpp26gf5rofi) **15**

[**Annexe B - Démonstrateur: Manuel de l’utilisateur**](#_7jxoe5nnpknh) **16**

[**Annexe C - Source et photos projet**](#_528mmiug5sou) **17**

# ACRONYMES ET DÉFINITIONS

|  |  |
| --- | --- |
| **Acronyme** | **Signification** |
| **PoC** | **P**roof **o**f **C**oncept |
| **IoT** | **I**nternet **o**f **T**hings |
| **EVM** | **E**thereum **V**irtual **M**achine |
| **DAPP** | **D**ecentralized **APP**lication |

|  |  |
| --- | --- |
| **Terme/Notion** | **Signification** |
| **Disruptif** | Terme désignant une rupture et nécessitant des changements majeurs dans les fondements / habitudes. Dans le cadre de la blockchain, le caractère disruptif correspond, entre autres, au modèle sans tiers de confiance. |
| **Immuable** | Qui demeure inchangé, n’est pas sujet au changement. |
|  |  |
|  |  |

# 

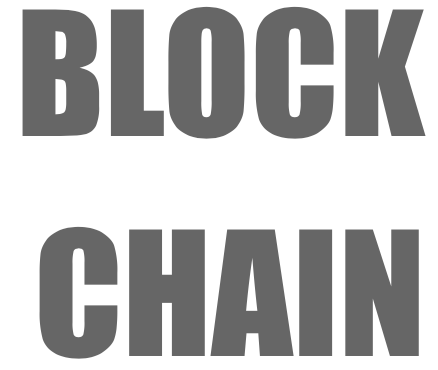
# RÉSUMÉ



Blockchain is a technology which was first revealed by the Bitcoin expansion. It now expands on its own without being stuck with the cryptocurrency image in order to explore new horizons. When redacting this report, numerous Proofs of Concept have been made, but the blockchain technology still struggles to find its way in companies for the industrialization.

We wrote a state-of-the-art report to recap the working of this technology. What are its basis ? What are its stakeholders ? What are its applications ? What are its pros and cons ?

We also made a demonstrator based on the Ethereum. Its principle is to use a blockchain to establish decentralized applications with the possibility to execute Smart Contracts. These can add external conditions on the transactions done on a blockchain. Our aim was to reuse the Ethereum framework in a project : we set up a roulette game involving the public attending our project presentation ; it allowed to point out the characteristics of a decentralized system like the blockchains with duplication of the information.

The conclusion is mitigated : on one hand, the blockchain technology offers a lot of possibilities and is disruptive in the process allowing to secure transactions without a trusted third party.

On the other hand, it doesn’t bring out a revolution as it doesn’t make impossible tasks with the current technologies.



La technologie blockchain a été introduite par le Bitcoin. Elle se démarque aujourd’hui de son image accolée aux crypto-monnaies afin d’explorer de nouveaux horizons. Au moment de la rédaction de ce rapport, nous sommes dans la phase où de nombreux Proof Of Concept (PoC) ont été réalisés mais la technologie blockchain a encore du mal à trouver son positionnement dans les entreprises pour une industrialisation.

Nous avons établi un état de l’art afin de faire le point sur le fonctionnement de cette technologie. Quels sont ses fondements? Ses acteurs? Ses applications? Ses points forts et ses faiblesses ?

Nous avons également établi un démonstrateur basé sur Ethereum. Le principe d’Ethereum est d’utiliser une blockchain afin de mettre en place des applications décentralisées et d’ajouter la possibilité d'exécuter des smart-contracts. Ceux-ci permettent d’ajouter des conditions sur les transactions exécutées sur une blockchain. Notre but était de réutiliser le framework de Ethereum dans un projet : nous avons mis en place une application de jeu de roulette en faisant participer l'auditoire durant la présentation dudit projet. Cela nous a permis de mettre en avant les caractéristiques de décentralisation et de duplication de l’information de cette technologie.

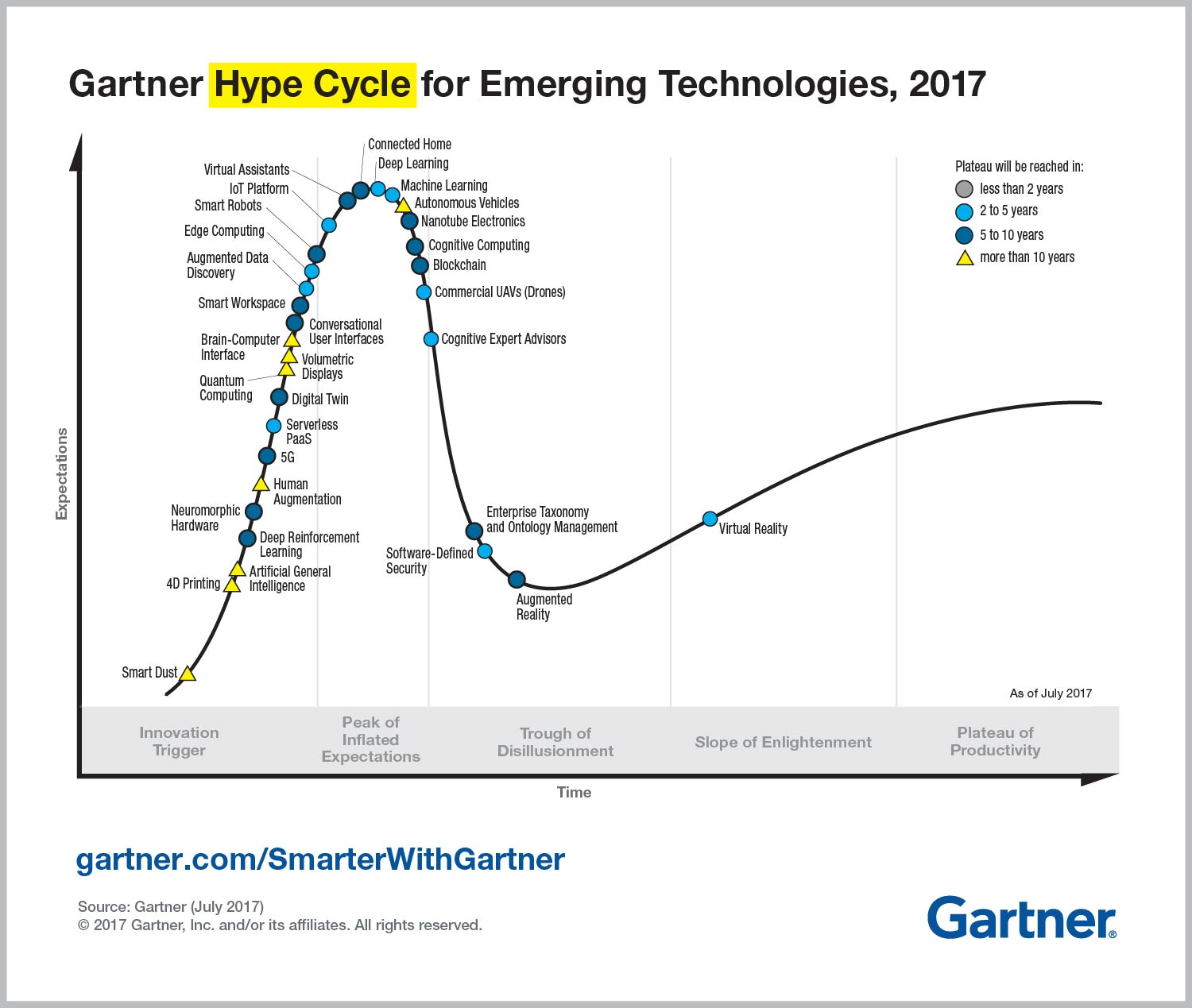
Nous sommes arrivés à des conclusions diverses. D’une part cette technologie offre de nombreux avantages et est disruptive dans le processus permettant de sécuriser les transactions sans tiers de confiance.

D’autre part elle n’apporte pas de révolution et ne réalise pas des tâches impossibles à réaliser avec les technologies d’aujourd’hui.

# 1 - INTRODUCTION

## 1.1 - Contexte

Bitcoin, blockchain, crypto-monnaie, disruption des banques, monnaie virtuelle, smart-contract, tiers de confiance etc…. Cette liste de mots n’a pas dû vous échapper si vous prêtez une oreille aux tendances technologiques du moment. Toutefois, que se cache t-il derrière ces termes ? Qu’est ce qu’une blockchain ? Quels sont les acteurs ? Comment pouvons-nous l’utiliser ? Quels sont les risques liés à la sécurité ? Quel est le potentiel ? C’est autant de questions auxquelles nous avons tenté de répondre au cours de ce projet innovation.



Comme nous pouvons l’observer sur cette courbe “hype” bien connue de chez *Gartner*, la technologie blockchain est tendance aujourd’hui. A l’image de sa position sur ce graphique, nous sommes situés au moment où la technologie est prometteuse. Cependant des interrogations subsistent (sécurité, potentiel etc..) avant de pouvoir la mettre en application.

De plus, ce projet s’inscrit dans le cadre d’une des huit chaires de l’université Grenoble INP visant à travailler sur des technologies émergentes. Celle-ci est nommée *Chaire Trust* et est focalisée sur la *confiance dans les systèmes de sûreté.*

*(source:http://www.grenoble-inp.fr/actualites/inauguration-de-la-chaire-trust-la-confiance-au-c-ur-de-votre-systeme-850877.kjsp)*.

## 1.2 - Objectifs

Le but de notre projet consiste à étudier la technologie blockchain dans sa dimension globale. Nous devons comprendre cette technologie afin de restituer les fondements dans un état de l’art. Ce travail permettra à un futur groupe d’étudiants de partir avec certaines bases afin d’établir un démonstrateur plus poussé.

Les pré-requis au niveau du démonstrateur n’étaient pas fixés. Nous avons eu le libre choix de la démonstration à mettre en place.

Nous avons décidé de créer un démonstrateur d’une application Ethereum. Ce choix a été motivé par plusieurs raisons:

* Ethereum est aujourd’hui la deuxième crypto-monnaie en terme de capitalisation
* Ethereum offre une nouvelle dimension de la blockchain en permettant de développer des applications décentralisées
* Ethereum permet de créer des smart contracts Turing complets
* Ethereum est aujourd’hui reconnu auprès des entreprises

# 2 - BLOCKCHAIN

## 2.1 - QU’EST-CE QU’UNE BLOCKCHAIN?

Au sens strict, une blockchain est constituée d’une séquence immuable de blocs, contenant des enregistrements de la preuve d’une information à un instant donné. Elle est immuable car un bloc *N* dépend de l’information du bloc *N-1*. Les blocs forment donc une chaîne. Si un bloc est modifié, la chaîne est cassée due à cette dépendance.

De manière à apporter de la sécurité et de la confiance dans une blockchain, plusieurs concepts sont associés

* **décentralisation** : d’après [[1]](#_xtkifb2qyj4k), l’architecture des systèmes répartis peut se classifier en 4 classes. De la classe 1 où une entité peut décider de tout ( centralisation avec un tiers de confiance), à la classe 4 où tous les noeuds sont autonomes. Sur une blockchain, tous les blocs sont répliqués sur tous les noeuds. Il s’agit plutôt d’un réseau de classe 3 où les **noeuds se mettent d’accord sur une décision**, ce qui mène au deuxième point :
* **consensus** : sans entité centrale qui décide/crée la chaine de blocs, un mécanisme permet à tous les noeuds du réseau de s’accorder pour synchroniser la même chaine de blocs. Pour obtenir le contrôle de la chaîne entière et modifier les blocs, il faudrait corrompre ce consensus. Ceci est d’autant plus difficile que le nombre de noeuds est élevé.
* **chiffrement asymétrique** : il est utilisé pour signer et chiffrer les échanges d’informations qui vont être inscrits dans les blocs.

### 2.1.1 Hash

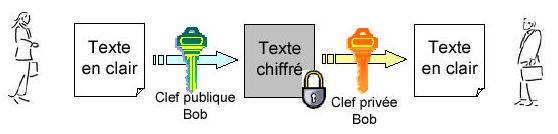
Le Hash permet d’affecter une empreinte digitale unique, de taille fixe, à un flot de données quelconque. Le changement d’un seul octet dans le flot de données changera totalement le Hash. Il est également impossible de retrouver les données d’entrée à partir du Hash. La figure 1 illustre le mécanisme de Hash. Dans la blockchain, le Hash correspond à la preuve de l'information énoncée ci-dessus.

*Figure 1: La fonction de hachage*

*(source:* [*http://www.ibibliotech.fr/4803-fonction-hachage*](http://www.ibibliotech.fr/4803-fonction-hachage)*)*

### 2.1.2 Chiffrement asymétrique

Pour communiquer des données sensibles entre un ou plusieurs interlocuteurs qui ne se connaissent pas, il est nécessaire de cacher le contenu des informations à des personnes mal intentionnées. Pour cela , il existe plusieurs types de chiffrement des données. Nous ne nous intéresserons ici qu’au chiffrement utilisé dans les blockchains, le chiffrement asymétrique (figure 2).



*Figure 2: Chiffrement asymétrique.*

*(source: http://igm.univ-mlv.fr/~dr/XPOSE2007/vma\_PKI/concepts\_de\_base.html)*

Avec ce type de chiffrement, chaque interlocuteur possède une **biclé**, soit un ensemble de 2 clés :

* une **clé publique** diffusée et connue par tous les autres interlocuteurs.
* une **clé privée** qui doit restée connue uniquement de son propriétaire.

Chaque clé possède deux fonctions.

* **Signature** : en signant une donnée ou un condensat de la donnée avec sa clé privée, un interlocuteur peut authentifier l’origine de cette donnée : il est possible de la déchiffrer avec la clé publique de cet interlocuteur, qui est connue de tous. Ainsi, tous les interlocuteurs savent de qui la donnée a été émise.
* **Chiffrement d’un secret** : en chiffrant une donnée avec la clé publique d’un interlocuteur, n’importe qui peut faire en sorte que ladite donnée ne soit lisible que par le propriétaire de la clé publique, qui va déchiffrer la donnée avec sa clé privée.

### 2.1.3 Transactions

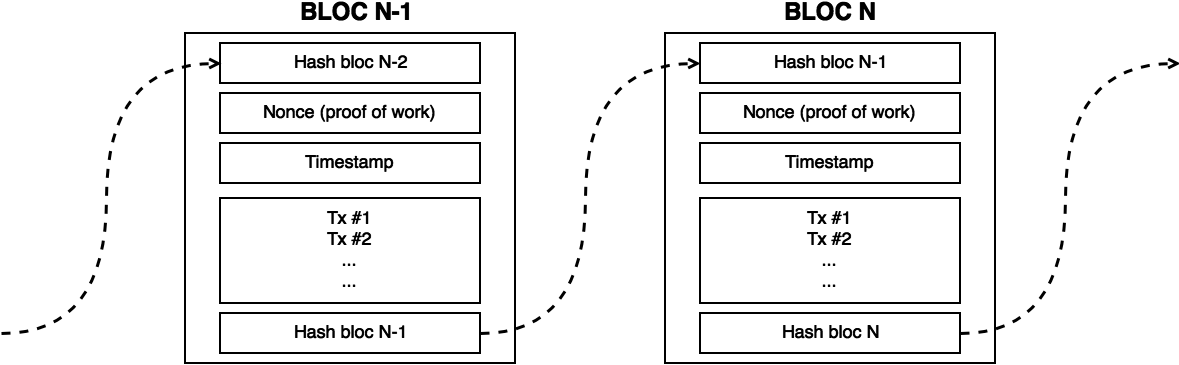
D’un point de vue informatique, une transaction est définie comme une opération de saisie ou de consultation d’une information *(source:* [*www.larousse.fr*](http://www.larousse.fr)*)*. Elle peut aussi faire référence à un changement d'état du réseau.

Lorsqu’un noeud du réseau veut saisir une opération sur une blockchain (envoyer de l’argent dans le cas d’une application de cryptomonnaie), les étapes suivantes vont être réalisées :

* l’information va être signée par l’utilisateur (cf chiffrement asymétrique)
* un Hash est généré à partir de ces données (preuve de l’information) et donne lieu à une transaction
* cette transaction est diffusée sur le réseau afin d'être ajouté à un bloc

### 2.1.4 Bloc

La figure 3 montre la structure d’un bloc :

[**](https://www.draw.io/#G1sZLhHcDFpKkVKkU0vYHByncNJDjNOXb8)

*Figure 3: Structure d’un bloc.*

La blockchain est un registre dont les pages sont les blocs qui contiennent une liste de transactions. Pour construire la chaine, le bloc contient aussi le Hash du bloc précédent, l’horodatage du bloc, et le Nonce. Ces données, avec les transactions, forment la source du Hash du bloc créé.

Le Nonce est un nombre particulier utilisé pour créer le bloc (ceci est vrai uniquement pour une blockchain utilisant la preuve de travail, comme Bitcoin et Ethereum) et est détaillé dans la section suivante.

Un bloc est créé à intervalle régulier même si il n’y a pas de transactions diffusées. En effet, il est nécessaire de construire des blocs et de faire grandir la chaîne afin de “sécuriser” les blocs précédents. L’intervalle de temps dépend de la blockchain.

### 2.1.5 Minage

Mais au fait, qui crée les blocs et comment ? Les noeuds qui créent les blocs sont appelés les *mineurs*.

Chaque blockchain peut décider de la stratégie pour créer les blocs, celle des blockchains Bitcoin et Ethereum utilise le principe de la preuve de travail utilisée.

La preuve de travail consiste à déterminer le Hash d’un bloc. Un paramètre spécifique, le Nonce, est associé aux données du bloc et modifié aléatoirement jusqu'à obtenir le Hash avec un nombre prédéfini (par le réseau) de “0” en entête. Ainsi, le bloc est créé et diffusé sur le réseau. Tous les noeuds vérifient que le Hash est valide.

L’annexe A résume les définitions apportées en illustrant le processus d’une transaction sur Bitcoin.

## 2.2 - ETHEREUM

Comme nous l’avons déjà mentionné , les blockchains ont été “démocratisées” par la cryptomonnaie Bitcoin. Par la suite, plusieurs applications ont été développées soit pour améliorer des points faibles du Bitcoin soit pour répondre à d’autres besoins.

Cependant chaque application déployait sa propre blockchain. Or, on sait que la force d’une blockchain réside dans son grand nombre de noeuds. L'idée de la fondation Ethereum consistait donc à développer une blockchain programmable, sur laquelle les applications peuvent être déployées sans se soucier du réseau qui est fourni par Ethereum.

Ethereum est aujourd’hui considéré comme stable et donc utilisé sur un réseau public (Internet) appelé *mainnet*. Il existe aussi des réseaux de test, et il est possible de mettre en place un réseau privé isolé d’Internet, soit pour du développement, soit pour des applications d’entreprises.

### 2.2.1 - L’EVM

La blockchain Ethereum est rendu programmable grâce à un élément clé: l’Ethereum Virtual Machine (EVM). Ethereum permet aux utilisateurs de créer leur propres opérations et applications. Mais ces opérations doivent être comprises par tous les noeuds du réseau ce qui est assuré par l’EVM.

Tous les noeuds du réseau opèrent l’EVM et exécutent les mêmes instructions. Ethereum peut donc être vu comme un ordinateur décentralisé. Cette redondance n’accélère pas les opérations, car au contraire Ethereum est beaucoup plus lent qu’un ordinateur classique. En effet, chaque exécution dans l’EVM est vue comme une transaction et doit donc être validée dans un bloc. Ainsi, Ethereum garantit un haut niveau de tolérance aux fautes, un stockage des données de la blockchain immuable et une résistance à la censure / corruption.

### 2.2.2 - Smart contract

Les *smart contracts* sont des programmes qui s'exécutent dans l’EVM présente dans chaque noeud de la blockchain Ethereum. Ils sont utilisés pour créer des Dapp (Decentralized application).

Un *smart contract* contient un solde (en Ether) et du (byte)code qui s'exécute dans l’EVM sur commande. Un *smart contract* ne s'exécute pas tout seul (sauf cas particulier ou le *smart contract* se peut se faire “réveiller”). On distingue deux types de commandes:

* les **appels** : commandes de lecture
* les **transactions** : commandes d'écriture qui changent l'état du réseau.

Quand un contrat reçoit une transaction, il exécute son code selon les paramètres d’entrée contenus dans la transaction. Le code du contrat est exécuté dans l’EVM de chaque noeud qui participe au réseau en vérifiant les nouveaux blocs.

**Remarques:** un *smart contract* est déployé par un *owner*. De plus, les *smart contracts* peuvent implémenter des fonctions permettant de “tuer” le contrat, c’est-à-dire, faire en sorte qu’il ne s'exécute plus. Or on a vu ci-dessus qu’un contrat possède un solde.

Qu’en est-il de ce solde si le *smart contract* est tué? Le solde revient à son *owner* ou à un compte défini dans le code du *smart contract*.

Il est donc important de note ici l’importance de lire le code d’un *smart contract* (dont on peut toujours connaître le code) avant de l'exécuter, au même titre que l’on lit un contrat classique écrit (souvent illisible, en police de caractère 6, avec des références croisées à n’en plus finir) avant de le signer.

D’un point de vue sécurité, les *smart contracts* présente une vulnérabilité importante dans Ethereum, comme on peut le voir dans cette article [[3]](#_xtkifb2qyj4k). Le code étant visible par tous, il est plus “facile” de trouver une faille dans le code d’un *smart contract*.

# 

# 

# 3 - DEMONSTRATEUR

## 3.1 - INTRODUCTION

Le démonstrateur permet d’appuyer l’explication de la blockchain à un public averti mais non expert dans la thématique. L’objectif principal est d’expliquer, de manière didactique, le fonctionnement d’une blockchain.

Pour cela, nos choix se sont concentrés autour de la volonté de faire participer l’auditoire afin que les concepts soient éventuellement mieux appréhendés mais surtout pour souligner la décentralisation. Nous avons donc décidé de déployer un réseau Ethereum “privé”, c’est à dire isolé d’Internet, plutôt que d’utiliser le réseau public qui requiert de posséder de la cryptomonnaie.

Bien que nous ayons démontré que la force et la sécurité d’une blockchain existe si elle contient un nombre élevé de noeuds sur son réseau, cette solution permet de mettre en place rapidement une blockchain basée sur Ethereum.

Concernant l’application, il s’agit d’un jeu de roulette. Les utilisateurs misent sur un nombre qui va sortir de manière aléatoire et gagnent ou perdent de l’argent suivant le résultat. Les utilisateurs peuvent aussi s’échanger de la cryptomonnaie et ainsi réaliser des transactions.

## 3.2 - ARCHITECTURE

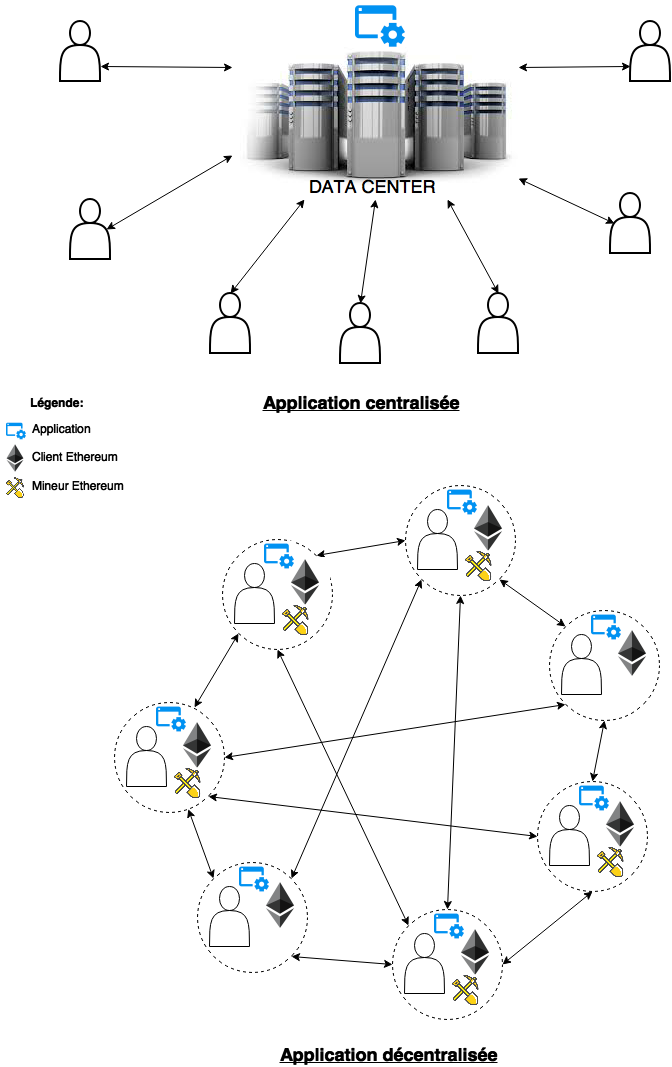
Tout d’abord, il est important de souligner l’importance et les avantages de la décentralisation apportée par la blockchain. La figure 4 montre la différence entre une application centralisée et décentralisée. Dans une architecture centralisée classique, 3 points principaux sont mis en cause à différents niveaux:

* **Technique :** bien qu’aujourd’hui les *data center* soient dotés de multiples redondances sur les serveurs et les alimentations, il peut arriver d’avoir une coupure totale de toutes les applications/services qui sont hébergé(e)s, comme la panne sur les serveurs OVH durant la journée du 9 novembre 2017.
* **Sécurité :** si une intrusion est effectuée sur un *data center*, la vie privée des utilisateurs de toutes les applications est potentiellement compromise. Dans le cas d’une attaque de type DoS (Denial of Service - déni de service), là encore, une coupure totale des applications/services hébergé(e)s peut avoir lieu.
* **Économique :** un *data center* offre une infrastructure payante. Plus son propriétaire est riche, plus il investit dans l’infrastructure, plus il héberge d’application/service, plus il gagne de l’argent.

Une blockchain apporte une alternative à ces 3 points. Du point de vue technique, les applications sont exécutées sur tous les noeuds du réseau. C’est pourquoi la force d’une blockchain réside dans le nombre : on est garanti qu’il y ait au moins un noeud qui exécute l’application/service. La notion *zero downtime* tant convoitée est donc assurée par la blockchain.

Concernant le 2ème point, la sécurité de la blockchain peut être menacée si le dernier bloc miné est toujours obtenu par le même noeud, ce qui est très compliqué à réaliser (et très couteux) pour ne pas dire *impossible*. Sans rentrer dans le détail, les attaques sur une blockchain ne sont possible seulement par manipulation du consensus du réseau, comme par exemple les attaques Sybil [[2]](#_xtkifb2qyj4k).

Enfin, du point de vue économique, de par sa décentralisation, le modèle de la blockchain redistribue le pouvoir de décision et les ressources .



En contrepartie, de part son modèle, Ethereum ne peut déployer toutes les applications. Un service de *streaming vidéo* par exemple n’est pas possible.

Ethereum, et la blockchain en général, n’est pas fait pour stocker des gros volumes de données.

De plus, les données de la blockchain sont synchronisées sur tous les noeuds ce qui pose un problème de scalabilité.

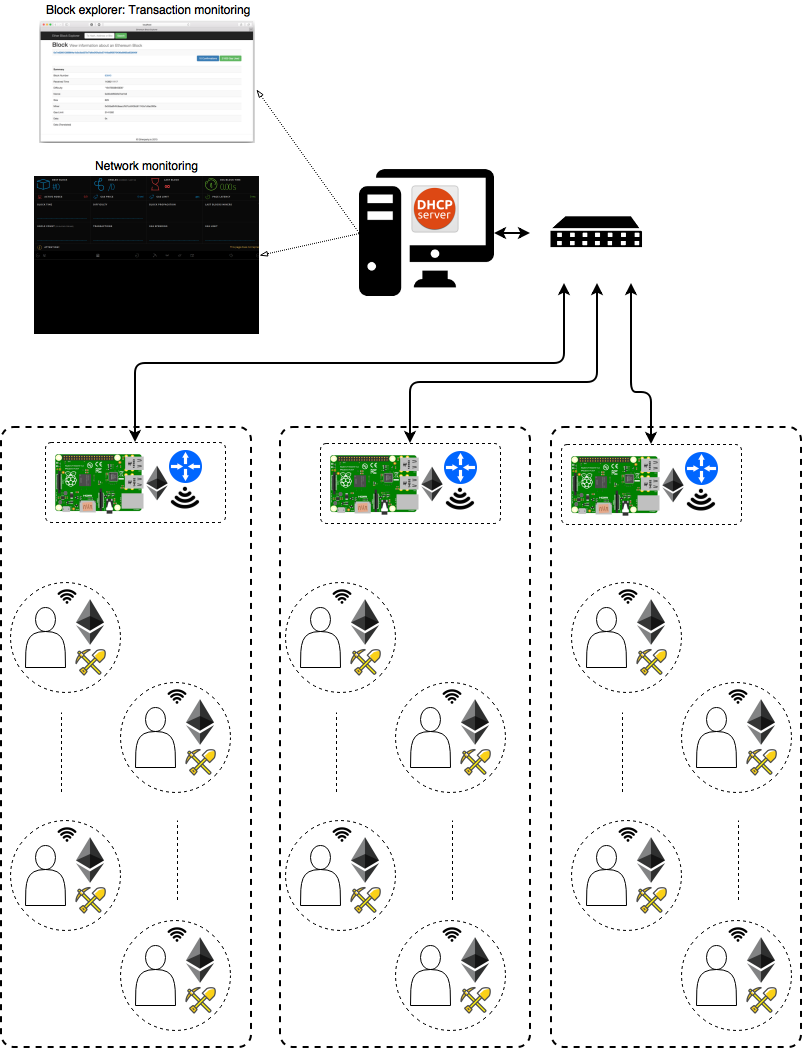
Aussi, des noeuds possédant une faible capacité de stockage ne peuvent pas synchroniser la blockchain (registre actuel d’Ethereum: 40GB).

Sur le schéma de l’application décentralisée de la figure 4, toutes les connexions pair-à-pair n’ont pas été représentées pour des questions de lisibilité. On remarque qu’il n’est pas nécessaire de *miner* pour pouvoir exécuter l’application.

|  |  |
| --- | --- |
|  | *Figure 4: Application décentralisée sur Ethereum VS application classique centralisée.* |

La figure 5 présente l’architecture du démonstrateur. Le réseau est constitué des éléments suivants:

* **Serveur DHCP :** permet d’assigner les adresses IP des noeuds du réseau local. Le serveur DHCP est installé sur une machine sur laquelle sont installés les outils de monitoring des statistiques du réseau et des blocs de la blockchain. Cette machine est aussi un noeud Ethereum du réseau.
* **Network switch :** permet de connecter les Raspberry Pi sur le réseau local.
* **Raspberry Pi :** elles sont utilisées en tant que Hotspot Wifi et routeur. Les 3 Raspberry Pi permettent de répartir les connections des utilisateurs sur le réseau local. L’utilisation des Raspberry Pi permet aussi d’apporter la dimension IoT au démonstrateur. Elles exécutent aussi un client Ethereum mais ne minent pas.
* **Utilisateurs :** se connectent au réseau local à travers une Raspberry Pi. Chaque utilisateur exécutent un client Ethereum et mine. Une interface web leur permet d’accéder à l’application et de réaliser des transactions de cryptomonnaie vers d’autres utilisateurs.

[](https://www.draw.io/#G1YYkEbXnYWREZo7rEnBLmEzGKvpL0UQqN)

*Figure 5: Architecture globale du démonstrateur.*

## 3.3 - APPLICATION

L’application est un jeu de roulette, où les joueurs (les utilisateurs) parient soit sur un numéro seul, soit sur la parité du nombre et mise une somme en Ether.

La banque (le contrat) récupère les mises et rembourse les paris gagnants.

Un lancer de la roulette est effectué par la banque. Les joueurs gagnants reçoivent leur mise multipliée selon le pari utilisé (x36 nombre seul, x2 pair ou impair).

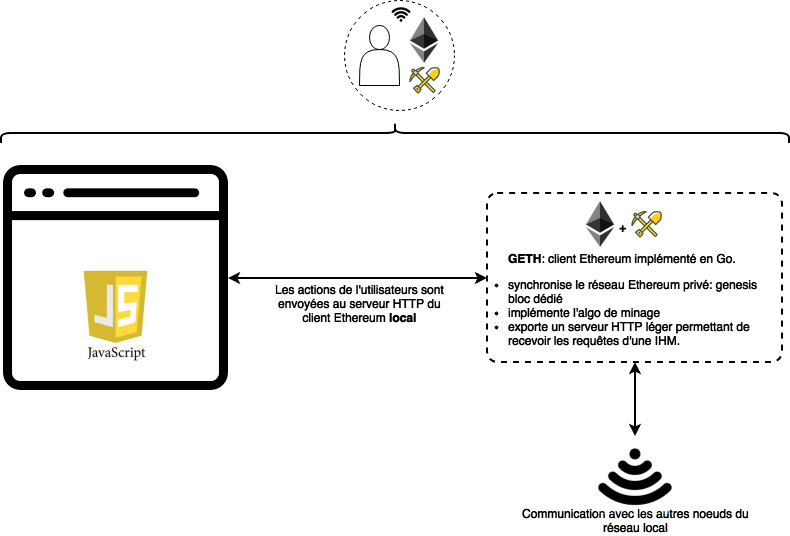
Toutes les actions de ce jeu (pari, mise, récupération des mises, remboursement des gains) sont des fonctions du contrat.

L’application a été packagée de manière à la déployer facilement pour la démonstration. Le package comprend:

* un script batch
* un client Ethereum (Geth)
* une IHM (page web)
* un front-end JavaScript faisant le lien entre l’IHM et le client Ethereum

Pour des raisons de contraintes de temps, le package a volontairement été défini pour supporter seulement la plateforme Windows 10.

La figure 6 détaille l’architecture logicielle de l’application:

[](https://www.draw.io/#G15Re7lzzTtBh6UVFlD7A-R5mW0i46cfx1)

*Figure 6: Architecture logicielle de l’application*

Le script batch non représenté sur la figure 6 permet d’installer tous les composants logiciels et de démarrer le noeud.

Un manuel de l’utilisateur peut être consulté en annexe B. Il permet de mettre en place le démonstrateur.

# 4. ANALYSE ET CONCLUSION

En pleine émergence, la blockchain semble être enfin détachée de son image négative qui la liait au Bitcoin. Les start-up sont florissantes et les grandes entreprises s’y intéressent fortement (ex: introduction en 2017 des “blockchains as a service” par Microsoft et IBM).

Etant donné l'évolution rapide de cette technologie et les décisions incertaines liées à sa méconnaissance et à sa méfiance, notre état de l’art sera surement à revoir d’ici un an.

**Décentralisation**

La blockchain apporte de la redondance et de la sécurité au réseau lorsque nous souhaitons traiter des données. Cela permet également de se passer d’un tier de confiance.

**Gouvernance**

Lors du déploiement d’une blockchain se pose le problème de gouvernance. Comment gérer un réseau si aucune entité centrale ne prend de décision?

**Est ce une technologie innovante?**

Son positionnement est atypique. Elle n’amène pas d’applications révolutionnaires. Par contre c’est lorsque que nous analysons les acteurs et le workflow sans tiers de confiance que la technologie apporte une dimension disruptive. (ex: banque vs bitcoin, aller plus loin dans la désintermédiarisation et se passer de Uber...)

**The Final thought(S)**

Nous avons pus développer chacuns des avis différents lors de notre étude.

|  |  |  |
| --- | --- | --- |
| **Mr génial**  C’est révolutionnaire!  Avec les crypto-monnaies, nous ne sommes plus tributaire des banques et pouvons effectuer des transactions sans tiers de confiance.  Les domaines d’applications sont très large: secteur musical, supply chain, cadastre, diplôme...  Décentralisation = sécurité, robustesse et confiance. Accord commun sans entité centrale qui m’impose ses règles, ses prix et sa mainmise sur ma vie privée. | **Mr taraude**  C’est intéressant mais…  Les crypto-monnaies pourraient être complémentaires aux monnaies fiduciaires et apporter de nouveaux usages.  La blockchain est sécurisé, mais il faut améliorer les applications tierces.  Améliorer la rapidité des temps de bloc?  La gouvernance est nécessaire pour certaines applications. | **Mr fumisterie**  Cela n’apporte rien!!!  Les crypto-monnaies ne reposent sur aucune valeur. De plus, cela me convient qu’une banque ou une entité centrale à laquelle je fais confiance gère mon compte.  Les applications ne sont pas révolutionnaires, on peut déjà faire ce qui est réalisé avec une blockchain.  Il y a des hacks tous les mois sur les plateformes de trading et de wallet.  Le réseau est lent et la scalabilité pose problème. |

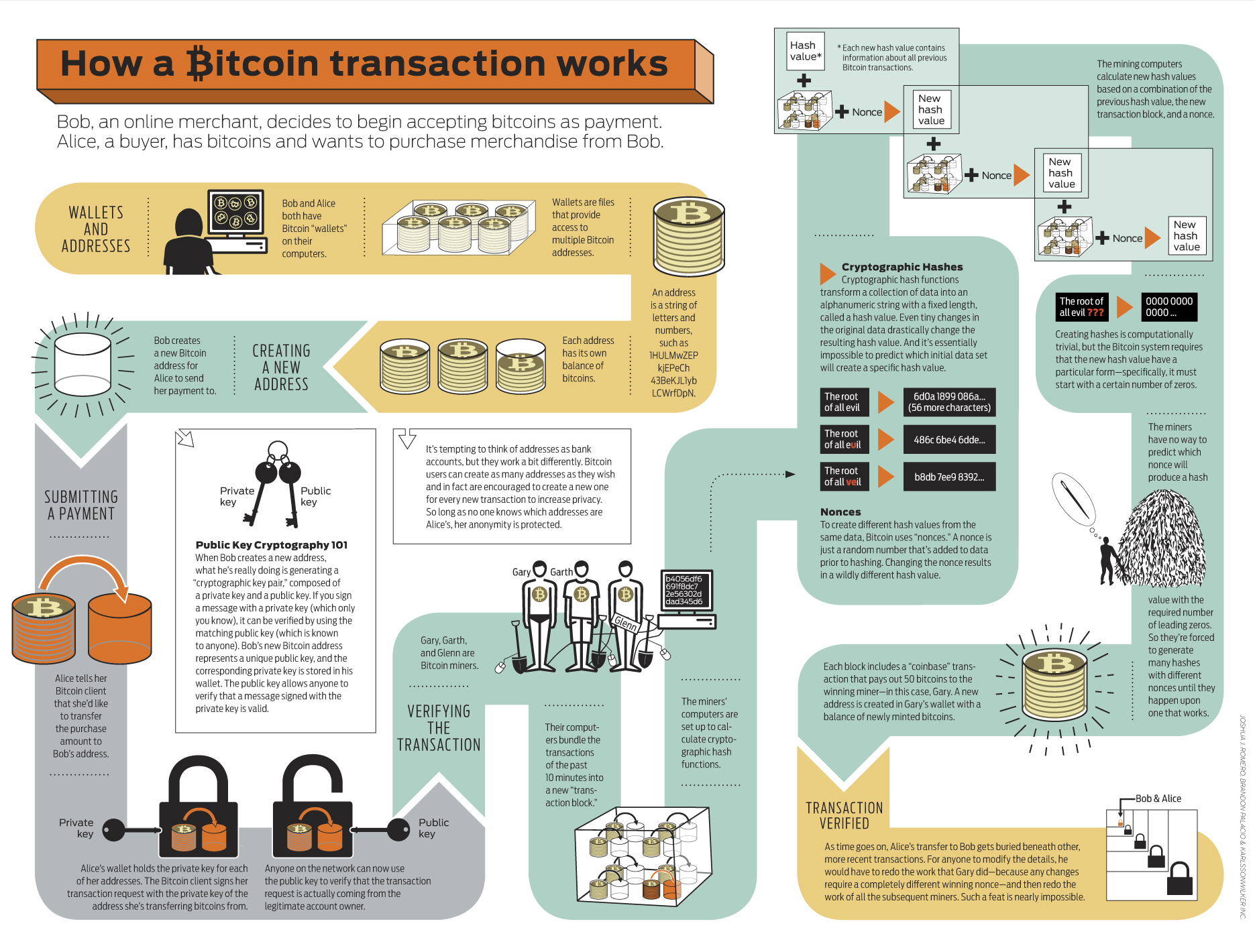
# RÉFÉRENCES

[1] Stéphane Bortzmeyer, *Centralisé, décentralisé, pair à pair, quels mots pour l’architecture des systèmes répartis ?*, AFNIC, 2015.

[2] John (DC) Douceur, *The Sybil Attack*, Microsoft, 2002.

[3] Haseeb Qureshi, *“A hacker stole $31M of Ether — how it happened, and what it means for Ethereum”*, [Online]. Disponible: <https://medium.freecodecamp.org/a-hacker-stole-31m-of-ether-how-it-happened-and-what-it-means-for-ethereum-9e5dc29e33ce>. Consulté le 07/12/2017.

# Annexe A - Illustration transaction Bitcoin



# Annexe B - Démonstrateur: Manuel de l’utilisateur

TODO

# 

# 

# Annexe C - Source et photos projet

**Source :**

A ajouter

**Photos :**

A ajouter