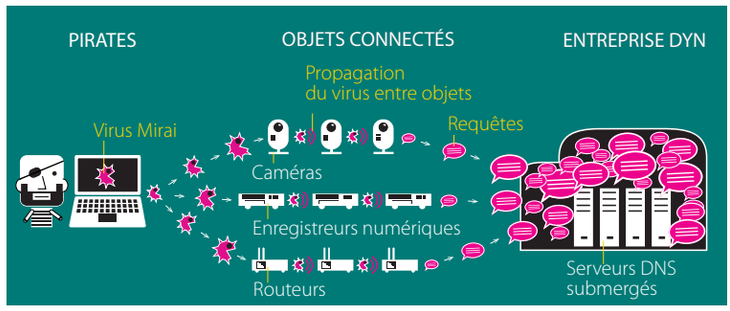
intégrer les concepts basiques de sécurité (confidentialité, intégrité, authenticité) dans l'IoT et en particulier dans les objets le constituant. La transformation de messages pour rendre leur contenu secret est probablement aussi vieille que l'écriture elle-même. Au VI^{e} siècle avant J.-C., les Grecs utilisaient la scytale, un bâton sur lequel était enroulée une fine bandelette sur laquelle le message était ensuite inscrit de manière longitudinale. Seule la bandelette était transportée par le messager : sans bâton du même diamètre, il était impossible de comprendre le message. Au fil des siècles, la cryptographie s'est développée, notamment par des techniques de transpositions, de permutations ou de substitutions. Dans le même temps, les méthodes de cryptanalyse, techniques de déchiffrement sans connaissance préalable de la clé secrète utilisée, ont elles aussi progressé.

L'avancée majeure en cryptographie vient sans conteste des travaux, dans les années 1970, de Whitfield Diffie et Martin Hellman qui ont introduit le concept de « cryptographie à clé publique », protocole d'échange de clés totalement sécurisé basé sur la notion de fonction à sens unique avec trappe ou *trapdoor one-way function*. C'est une fonction qui se calcule facilement dans un sens, mais qui est mathématiquement impossible à inverser si l'on ne connaît pas un secret (trappe), bien que cette fonction soit connue de tous. La fonction, ou clé publique, sert à chiffrer le message. Connue d'un cercle restreint d'utilisateurs, la trappe ou clé privée est nécessaire pour déchiffrer le message. L'exemple le plus connu de cryptosystème à clé publique est apparu en 1978, inventé par Rivest, Shamir et Adleman d'où le nom de RSA. Il s'appuie sur la difficulté de factoriser les grands nombres entiers.

Depuis, bien d'autres systèmes de chiffrement à clé publique ont été développés. Celui qui est aujourd'hui en passe de remplacer le RSA est un système de chiffrement fondé sur des problèmes de calcul de logarithmes discrets. En 1985, Miller et Koblitz ont posé indépendamment les bases d'une cryptographie à clé publique à partir de courbes elliptiques (*Elliptic Curve Cryptography* – ECC). La sécurité de ces systèmes repose sur le problème du logarithme discret sur courbes elliptiques. L'algorithme connu comme étant le plus efficace pour résoudre un tel problème est à temps de calcul exponentiel, contrairement au

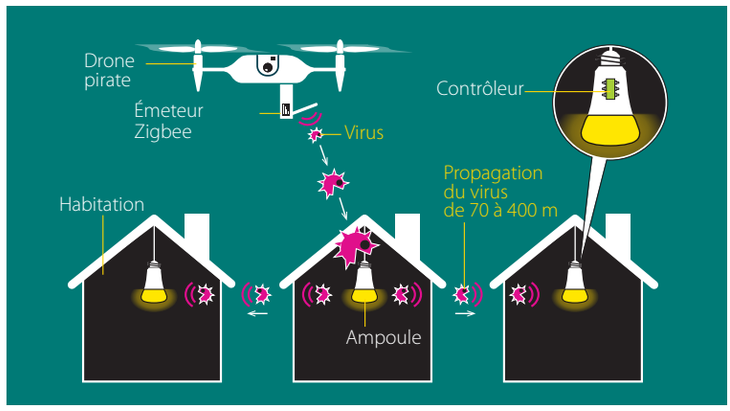
Les objets connectés, cibles des pirates

Livrés avec des mots de passe par défaut, les objets connectés sont une cible facile pour Mirai. Ce virus a servi lors de l'attaque du 21 octobre 2016 à les enrôler dans un réseau contrôlable à distance : un botnet. Ils ont été mobilisés par les pirates pour envoyer un déluge de requêtes simultanées au service DNS de l'entreprise DYN, dédié à la traduction des noms de domaine (www.industrie-techno.com) en adresse IP. Résultat : submergé, le serveur n'a plus été capable de jouer son rôle d'aiguilleur, interdisant l'accès à plusieurs sites grand public.



Piratage d'une ampoule connectée

Des chercheurs ont prouvé qu'un drone pouvait émettre un signal Zigbee contenant un virus capable d'infecter des ampoules connectées, qui se contaminent alors de proche en proche. En survolant les différents quartiers d'une ville, infecter 15 000 ampoules suffit à prendre le contrôle de toutes celles d'une ville comme Paris, sur une superficie de 105 km^{2}. On peut alors commander à distance l'allumage des ampoules, ou encore les utiliser comme un botnet pour générer des requêtes Internet en cascade et perturber ainsi le fonctionnement normal des serveurs.



RSA pour lequel il existe des algorithmes à temps de calcul sous-exponentiel. Ainsi, à niveau de sécurité équivalent, ECC requiert des clés pouvant être jusqu'à dix fois plus petites que le RSA. Ceci le rend très attractif pour les applications qui possèdent des ressources de calcul limitées (voir encadré « Principe du chiffrement asymétrique »).

Vulnérabilités : le chiffrement ne suffit pas

La cryptographie a permis d'introduire un niveau de sécurité élevé dans les systèmes de communication et services associés. La sécurité qui en découle repose essentiellement sur la capacité des circuits intégrés utilisés à garder secrètes les clés de (dé)chiffrement. Or les techniques d'attaque de systèmes physiques cherchant à extraire ces clés de chiffrement ne cessent de progresser.

Les attaques intrusives visent à observer visuellement, ou électriquement, le composant ou le système chargé d'assurer la sécurité. Chaque niveau peut être mis à nu pour étudier et recréer le schéma électrique du composant et identifier les cellules de base afin, par exemple, de reconstituer des fonctionnalités cachées ou secrètes (voir encadré « Principe d'une attaque par rétroconception »). Ce type d'attaque est très complexe et souvent coûteux, notamment s'il doit s'appliquer à l'ensemble d'un circuit. C'est pour cela qu'il est plutôt ciblé, pour découvrir des contre-mesures (protection active, couche anti-intrusion, générateur de nombres aléatoires) ou identifier des sous-ensembles critiques du composant sur lesquels on pourra focaliser ses efforts.

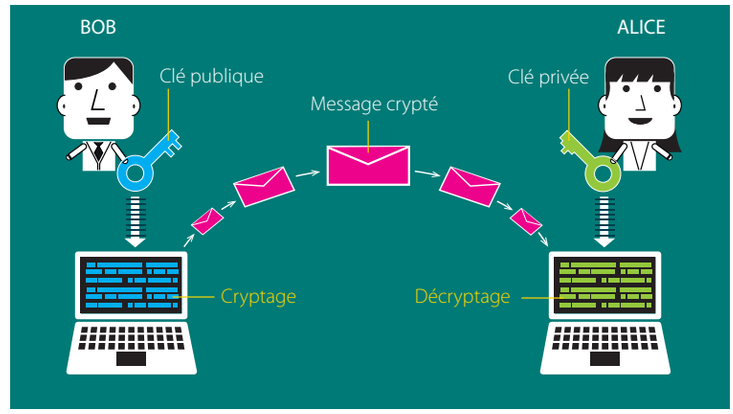
S'il est possible de créer un accès à une piste du circuit électronique, on pourra récupérer les signaux transitant sur cette piste et en particulier des données secrètes. De telles attaques sont d'autant moins faciles que la géométrie des circuits se miniaturise d'année en année. Car il faut être capable d'acquérir, en une seule fois, les signaux de plusieurs pistes – par exemple l'ensemble des voies d'un bus de communication – ou de les mesurer un par un, puis de les resynchroniser.

Les attaques intrusives permettent également de modifier le fonctionnement normal du circuit, en coupant des pistes internes ou en en créant de nouvelles, pour désactiver des contre-mesures et priver le circuit de ses protections afin de conduire d'autres types d'attaque. Ces approches sont d'un niveau de complexité très élevé sur les circuits actuels et ne sont accessibles qu'à certains laboratoires dotés d'équipements et de compétences très spécialisés.

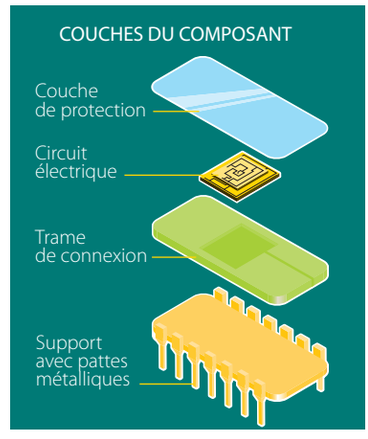
D'autres types d'attaques, non intrusives, ne modifient pas le composant. Il s'agit d'analyser son fonctionnement à l'aide de données physiques observables – les canaux auxiliaires – telles que la consommation électrique, la durée d'exécution ou le rayonnement électromagnétique émis : autant de sources d'information liées aux données traitées par le circuit qui permettent de mener les attaques par canaux auxiliaires. Ainsi, on pourra observer les fuites de courant intrinsèques des composants électroniques ou exploiter le fait que les différentes parties d'un circuit ne consomment pas la même énergie pour déterminer passivement les opérations exécutées par le circuit (par exemple, les algorithmes

Principe du chiffrement asymétrique

Les systèmes à chiffrement asymétrique, comme le plus connu d'entre eux, appelé RSA, reposent sur l'utilisation de deux clés, l'une publique et accessible à tous et l'autre privée, qui ne sert qu'à les déchiffrer. Tout se passe comme si Alice donnait à Bob accès à un coffre-fort où il peut déposer un message avant d'en claquer la porte qu'elle seule pourra ouvrir. La cryptographie sur les courbes elliptiques (ECC) repose sur le même principe, mais avec des clés plus courtes, utilisant moins d'énergie pour la transmission, et permettant d'obtenir une meilleure sécurité à coût équivalent.



Principe d'une attaque par rétroconception



L'auteur d'une attaque par rétroconception retire les éléments qui protègent la puce et extrait cette dernière. Ce n'est qu'ensuite qu'il étudie le circuit électronique après avoir désactivé les éléments qui jouent le rôle de cage de Faraday, empêchant la surveillance des champs électromagnétiques émis par le circuit. L'attaquant peut alors connecter des fils ou couper des pistes à des endroits bien choisis pour casser le secret du composant.

cryptographiques exécutés) ou retrouver, à travers des traitements statistiques, les données manipulées (par exemple, les clés cryptographiques) (voir encadré « Principe d'une attaque par observation des canaux auxiliaires »).

Une troisième méthode d'attaque, l'injection de fautes, induit un comportement anormal dans un circuit à l'aide de stimuli électriques ou physiques ; par exemple en modifiant la température du composant, en l'exposant à des rayonnements X, UV ou visibles, en altérant son alimentation électrique, en modifiant la fréquence de fonctionnement de son horloge, etc. Les stratégies d'attaque utilisant les perturbations sont aussi variées que les effets potentiels. Elles reposent sur l'analyse du circuit, de ce qu'il est supposé faire et de ses protections. De telles attaques peuvent permettre de repérer l'emplacement de fonctions critiques du circuit, de corrompre l'exécution d'un mécanisme de sécurité (par exemple contourner la vérification d'un compteur d'essais de code PIN) ou de retrouver des clés cryptographiques secrètes en faisant de la cryptanalyse différentielle à partir d'un message chiffré correctement et d'un autre chiffré de manière erronée à cause de l'injection de faute (voir encadré « Principe d'une attaque par injection de faute »).

Ces attaques physiques permettent d'exploiter les faiblesses inhérentes aux circuits intégrés actuels (fuite de courant ou sensibilité aux fautes) pour affaiblir l'implémentation des algorithmes cryptographiques pourtant mathématiquement robustes.

Prévention : une conception adaptée pour prévenir les attaques

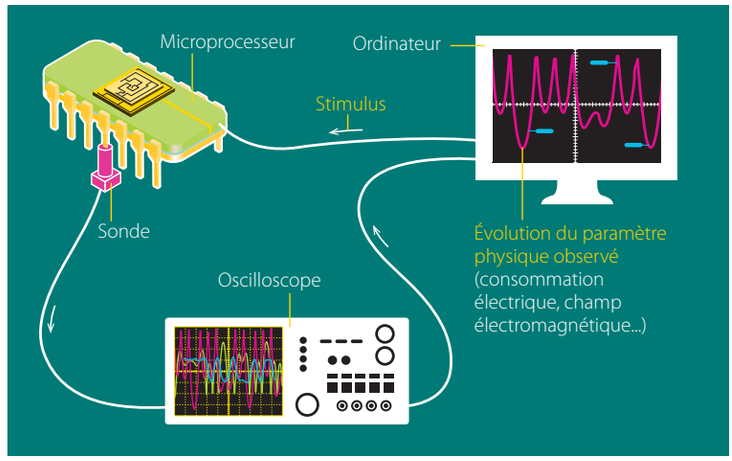
Quelle que soit la sophistication d'un circuit, des vulnérabilités sont susceptibles d'apparaître. Pour garantir la sécurité d'un système, il est nécessaire de comprendre les mécanismes mis en jeu lors d'une attaque. De même, puisque les concepteurs cherchent à réduire la taille des circuits intégrés pour gagner en performance et en autonomie, il convient d'étudier l'impact de l'utilisation de technologies de fabrication avancées sur la sécurité des systèmes.

Pour contrer les attaques physiques, les mesures de protection doivent être intégrées au plus tôt lors de la conception du circuit dont le cycle de vie doit être pleinement maîtrisé, et ce, pas uniquement, comme c'est le cas aujourd'hui, pour les circuits des « cartes à puce » (voir encadré « Architecture d'une puce sécurisée »). Ces contre-mesures, souvent étudiées pour limiter leur impact sur la surface, la consommation et les performances du système, doivent faire face à des attaques dont l'évolution est particulièrement rapide. Ainsi, tout un arsenal de contre-mesures doit être disponible. Outre les contre-mesures logiques, qui consistent à mettre en place des protocoles et des algorithmes de protection des données, on distingue deux types de contre-mesures, comportementales et physiques.

Les contre-mesures comportementales, également appelées structurelles, sont multiples. Ainsi, les signaux susceptibles d'être observés lors d'une

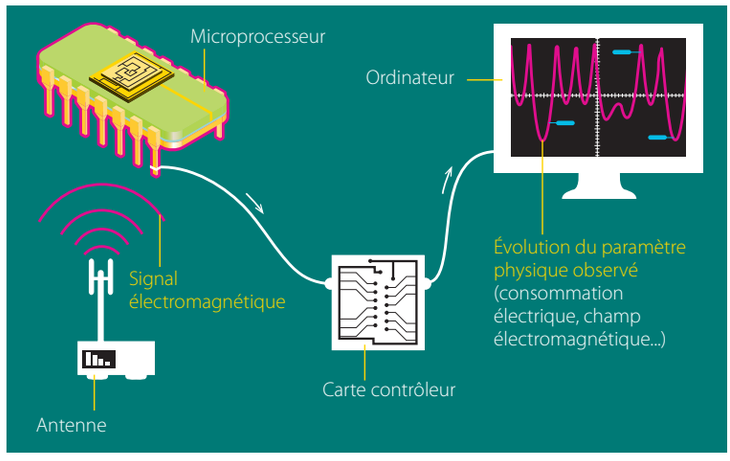
Principe d'une attaque par observation des canaux auxiliaires

Le pirate connecte le microprocesseur à un ordinateur qui envoie un stimulus pour le faire fonctionner et à un oscilloscope. La forme des courbes d'évolution du paramètre physique observé donne des informations sur le fonctionnement du circuit. Par exemple : la consommation électrique d'un bit « 1 » est supérieure à celle d'un bit « 0 ».



Principe d'une attaque par injection de faute

L'attaquant connecte le microprocesseur à un ordinateur et à une carte contrôleur pour la faire fonctionner, puis envoie un signal (électromagnétique, laser ou lumineux) sur le circuit pour provoquer un dysfonctionnement. L'étude des courbes de consommation de courant obtenues permet de remonter au secret.



attaque peuvent être dégradés en utilisant des techniques de désynchronisation (ajout de cycles d'exécution fictifs, utilisation d'une horloge instable variable, etc.). Une logique complémentée peut aussi être envisagée : plus chère, cette méthode associe à chaque bit d'information (0 ou 1) son complémentaire (1 ou 0). De cette manière, on s'assure que la consommation électrique du microprocesseur est indépendante des données traitées. Une très grande variété de contre-mesures logicielles est disponible, qui s'étale du niveau applicatif au niveau algorithmique. On peut tout de même les répertorier en deux catégories principales : le masquage des données, ou « randomisation », consiste à rajouter de l'aléa aux données manipulées pour décorréler leurs valeurs des canaux auxiliaires du circuit. Une technique qui requiert beaucoup d'efforts d'optimisation, car elle augmente le temps de calcul et la mémoire utilisée pour stocker les données. La seconde technique, baptisée obfuscation, tente de masquer le lien entre les données et la consommation électrique qu'elles induisent. Concrètement, cela signifie qu'on s'arrange pour que chaque opération requière approximativement la même consommation ou pour que celle-ci soit aléatoire. Mais, en pratique, il reste une certaine dépendance entre les données et la consommation associée.

Les contre-mesures physiques visent à modifier l'architecture physique du circuit pour éviter, par exemple, une relecture des données stockées en mémoire (brouillage ou chiffrement des données) ou faciliter la détection de fautes malveillantes (redondances spatiales et temporelles des circuits et données manipulées).

Perspectives : évolution des outils cryptographiques

L'émergence de l'IoT est aujourd'hui possible grâce aux nombreux protocoles de communication (Zigbee, Sigfox, Lora) développés pour les marchés concernés et à l'apparition d'architectures de processeurs de basse consommation, comme les puces ARM. Cette firme, qui travaille depuis les années 1990 avec Apple, propose des architectures de composants Cortex-M qui ont apporté des progrès majeurs. Néanmoins, les applications complexes utilisent souvent aujourd'hui des architectures Cortex-A ou classiques (x86, etc.). Le défi est de les porter sur des puces à basse ou très basse consommation comme les Cortex-M.

Les objets connectés n'échappent pas aux contraintes rencontrées pour toutes les applications embarquées, à savoir la capacité mémoire disponible, la puissance de calcul et la consommation électrique. Un triple obstacle pour la cryptographie, qui consomme beaucoup de ces trois ressources.

La sécurité dans l'IoT nécessite d'assurer la confidentialité, l'authenticité, l'intégrité, la disponibilité et la non-répudiation des données, sans compter la préservation de la vie privée. La plupart de ces aspects reposent sur l'utilisation de la cryptographie, avec deux algorithmes qui s'imposent aujourd'hui dans l'IoT : l'AES pour le chiffrement ou l'authentification symétrique et les courbes elliptiques pour le chiffrement asymétrique ou la signature. Or, ces algorithmes ne permettent pas de répondre pleinement aux exigences de performance, de consommation et d'empreinte mémoire de l'IoT. Ainsi, des travaux scientifiques et de standardisation mettent l'accent sur de nouveaux algorithmes dits légers (*lightweight cryptography*), qui se doivent en plus d'être intrinsèquement résistants aux attaques physiques décrites précédemment. De plus, en janvier 2015, la Commission européenne a émis auprès de ses organismes un mandat sur la protection des données privées. Elle demande que les données produites par les capteurs soient accessibles par l'utilisateur, qui doit avoir un droit de regard sur ses données personnelles. Un groupe de travail de l'Institut européen des normes de télécommunications (Etsi) travaille aujourd'hui sur des protocoles de type Abac (*Attribute based access control*) pour verrouiller l'accès aux données personnelles récoltées par les objets connectés.

Pour garantir la confidentialité de bout en bout (c'est-à-dire depuis le nœud qui collecte les données jusqu'au serveur du *cloud* qui les traite), la cryptographie homomorphe est une voie prometteuse. Elle permet d'effectuer tout type de calculs sur des données chiffrées sans avoir à les déchiffrer. Ce nouvel outil n'en est qu'à ses balbutiements, car il est à ce jour encore trop lent pour un déploiement à grande échelle.

Pour compliquer encore le tout, l'essor de l'informatique quantique est à prendre en compte pour des objets intelligents déployés dans nos maisons, nos usines et nos villes. La puissance de calcul de ces futurs ordinateurs pouvant être une vraie menace quant à la sécurité des schémas de type RSA ou ECC, des algorithmes appelés *post quantum* ou *quantum safe* sont aussi en cours d'étude, notamment quant à leur intégration dans les infrastructures IoT. Pour répondre aux menaces persistantes sur les circuits intégrés du marché, les concepteurs se doivent d'y intégrer dès la phase de conception les contre-mesures nécessaires et adaptées aux attaques physiques. Cette intégration de contre-mesures doit avoir un impact limité sur le coût final du système et ne pas retarder la commercialisation des produits. Le cycle de développement du système protégé doit donc être court. Par ailleurs, il faut s'assurer que les protections apportées soient les mêmes d'un circuit à l'autre (critère de reproductibilité). À cela s'ajoute

le critère crucial de testabilité des contre-mesures face à des attaques actives ou passives, qui peuvent être coûteuses en durée de test. Enfin, il convient de s'assurer de l'impact du vieillissement sur les performances et la sécurité du circuit.

Prétendre concevoir un circuit avec une sécurité absolue serait utopique, car les technologies évoluent constamment et les attaquants sont de plus en plus inventifs et ingénieux. Cependant, la compétition qui s'exerce entre des technologies de défense de plus en plus sophistiquées et des attaques de plus en plus agressives confirme ce que les cryptographes connaissent depuis très longtemps : la complexité d'un cryptosystème peut réclamer des efforts tels pour le casser que cela devient complètement dissuasif. Ce qui serait plus pragmatique à long terme pour l'industrie naissante de l'IoT, c'est la mise en place de standardisations tant au niveau de la conception sécurisée des circuits qu'au niveau de la mesure de leur résistance aux attaques. Cette approche de standardisation via des référentiels de sécurité a fait ses preuves pour la carte à puce et devrait pouvoir être adaptée et déployée pour l'IoT. La survie de ces systèmes connectés et celle de nos sociétés qui en sont de plus en plus tributaires pourraient en dépendre. ■

Architecture d'une puce sécurisée

Les cartes à puce utilisées il y a une quinzaine d'années avaient un premier niveau de sécurisation grâce à l'intégration d'un générateur de nombres aléatoires, capable de fournir des clés de chiffrement incluant un aléa pour garantir leur sécurité et d'un accélérateur de calculs cryptographiques dédié aux opérations liées au cryptage du message issu du processeur à l'aide de ces clés. Les puces sécurisées actuelles ont d'autres protections, qui interdisent notamment de distinguer les différents blocs que l'on peut voir apparaître sur cette image plus ancienne.