UNIVERSIDADE DA BEIRA INTERIOR
Engenharia

Mapeamento de Requisitos de Segurança à Tecnologia na Internet das Coisas

Moser Zeferino Vicente José

Dissertação para obtenção do Grau de Mestre em
Engenharia Informática
(2º ciclo de estudos)

Orientador: Prof. Doutor Pedro Ricardo Morais Inácio

Covilhã, Outubro de 2018

Agradecimentos

Primeiro agradecer a Deus pela vida, saúde e, por ter-me concedido a oportunidade de terminar mais um ciclo. Aos meus pais por acompanharem o meu trajeto e por estarem sempre presente na minha educação e formação. De igual modo, quero agradecer ao Professor Doutor Pedro Ricardo Morais Inácio pela incansável dedicação em orientar-me na elaboração desta dissertação.

Aos meus colegas e amigos, em particular a Alma Luzendo, a Fátima Dantas, o Jusualdo Figueira, o Gaspar António, o João Mota e o Evaldo Chindele, por apoiarem-me nos momentos mais difíceis aquando da nossa estadia nesta cidade, mais propriamente nos momentos de desânimo e nas noites mal dormidas. Agradecer, acima de tudo, por todos os momentos de alegria, confraternização e boa disposição. Agradecer também ao Bernardo Sequeiros e ao Musa Samaila pela disponibilidade em me ajudarem nos momentos de dúvidas que tive durante a realização deste documento.

Deixo um agradecimento muito especial a Judith Bizarro, Victor, Susana, Teresa, Zé Manel, Madalena por me apoiarem sempre espiritualmente e todas as vezes que precisava de aconchego. A todos que indiretamente participaram na realização deste documento quero expressar a minha gratidão por isso.

Não poderia deixar de agradecer à Joana Farias pela amizade, carinho, simplicidade e pela disponibilidade em me ajudar nos momentos difíceis aquando da realização deste trabalho e sobre tudo pela motivação que me dava.

À minha família em geral, quero agradecer pelo carinho e, apoio prestado nos momentos difíceis e importantes mesmo a vários quilómetros de distância.

Por último e não menos importante quero agradecer a todos os professores que tive desde o primeiro dia de aulas e, em especial, ao corpo docente da Universidade da Beira Interior (UBI).

A todos o meu muito obrigado.

Resumo

Devido ao rápido avanço tecnológico, a segurança e a privacidade têm sido assuntos de debates nos últimos anos, tanto por parte da comunidade industrial e científica, como pelos utilizadores finais. A partilha de informação através de métodos tecnológicos tem vindo a crescer na última década, devido também ao rápido crescimento e penetração destes meios. Uma das áreas que maior crescimento tem apresentado é conhecida como a Internet das Coisas (IdC), que engloba tecnologia que permite conectar vários objetos físicos e inteligentes com poder computacional reduzido na Internet, permitindo a partilha de informações entre si. Atualmente a IdC já desempenha um papel importante em muitas áreas chave da sociedade (e.g., transportes, saúde, e agricultura), sobretudo por permitir, de forma cómoda, extrair e partilhar informações referentes ao estado do ambiente. No entanto, vários problemas e desafios relacionados com a segurança têm surgido e demonstrado que a IdC carece de uma intervenção estruturante nessa dimensão, começando pelo básico.

A IdC está num estado prematuro em termos de segurança e privacidade. Não existe uma abordagem geral (ou até preocupação) em relação à segurança, e não existe um mecanismo padronizado para a proteção dos dados e dispositivos. O *hype* associado a novos sistemas da IdC degenera num *time-to-market* reduzido que beneficia funcionalidade e prejudica engenharia de segurança. Por outro lado, especificações de processamento e memória mais modestas para muitos dos dispositivos da IdC criam desafios adicionais ao desenho e à integração de mecanismos de segurança; o facto de muitos dispositivos estarem fisicamente acessíveis, sem controlo rígido de acesso, bem como o facto de tecnologias sem-fios muitas vezes não cifradas ou sem garantia de integridade serem usadas nas comunicações da IdC facilita sobremaneira ataques a dispositivos IdC por parte de entidades maliciosas.

Esta dissertação começa por analisar, de forma abrangente, alguns estudos já realizados na área da segurança para a IdC, comparando-os e usando-os como motivação para um estudo sobre a performance de mecanismos de segurança em plataformas de desenvolvimento atuais da IdC, especificamente num Raspberry Pi 3, assumindo que este conhecimento é importante para o desenho de sistemas e software para a IdC. A dissertação contém os resultados da performance para algumas funções da biblioteca *OpenSSL*, incluindo as funções de *hash* e cifra mais populares atualmente, colocados em perspetiva contra os resultados obtidos para um computador de secretária. É ainda apresentado um conjunto de testes de desempenho relacionadas aos mecanismos de segurança existentes para sistemas computacionais (e.g., *firewall* e sistema de detenção de intrusão).

Esta dissertação apresenta também uma primeira abordagem para o mapeamento entre requisitos e mecanismos de segurança no contexto específico da IdC, com objetivo de encontrar os requisitos que melhor se adaptam ao ambiente da IdC, conhecendo as suas limitações, em particular a de segurança. No final, é apresentado um protótipo de uma ferramenta para prova de conceito e que dá utilidade a este mapeamento.

Palavras-chave

Internet das Coisas, Mapeamento, Mecanismos e Controlos de Segurança, Objetos Inteligentes, Plataformas para a IdC, Raspberry Pi 3, Requisitos de Segurança.

Abstract

Due to a very fast paced rythm in technology advancement, security and privacy have been subjects of debate in recent years, both by the industry and scientific community, and by end users. The sharing of information through technological means has been growing in the last decade, also due to the rapid growth and penetration of technology in everyday human life. One of the fastest growing areas is known as *Internet of Things* (IoT), which encompasses technology that allows one to connect multiple physical and intelligent objects with reduced computing power to the Internet, allowing sharing information with each other. Presently, IoT already plays an important role in many key areas of society (e.g., transport, health, and agriculture), in particular by enabling it to conveniently extract and share information on the state of the environment. However, several security-related problems and challenges have arisen in the last years, clearly demonstrating that IoT lacks a structuring intervention in this dimension, starting from the basics.

IoT is in a premature state in terms of security and privacy. There is no general approach (or even concern) to security, and there is no standardized mechanism for protecting data and devices. The *hype* associated with new IoT systems often degenerates into a reduced *time-to-market* that benefits functionality and prejudices security engineering. On the other hand, more modest processing and memory specifications for many of the IoT devices create additional challenges in designing and integrating security mechanisms; the fact that many devices are physically accessible without rigid access control as well as the fact that often unencrypted or unsecured wireless technologies are used in the communications of IoT greatly facilitates attacks on such devices by malicious entities.

This dissertation begins by analyzing, in a comprehensive manner, some studies already carried out in the field of security for IoT, comparing them and using them as motivation for a study on the performance of security mechanisms in current development platforms for the IoT, specifically in a Raspberry Pi 3, assuming that this knowledge is important for the design of secure IoT systems and software. The dissertation contains the performance results for some OpenSSL library functions, including the most popular hash and cipher functions currently used. These results are compared against the ones obtained for a desktop computer. Also presented is a set of performance tests concerning existing security mechanisms for computational systems (e.g., firewall and intrusion detection system).

This dissertation also presents a first approach to the mapping between requirements and security mechanisms in the specific context of IoT, in order to find the requirements that best fit the IoT environment, knowing its limitations, in particular security. In the end, a prototype of a proof-of-concept tool, which shows how this mapping can be useful in practice, is presented.

Keywords

Internet of Things, Mapping, Security Mechanisms and Controls, Smart Devices, Platforms for the IoT, Raspberry Pi 3, Security Requirements.

Conteúdo

Lista de Figuras

Lista de Tabelas

Lista de Acrónimos

3DES	*Triple Data Encryption Standard*
6LoWPAN	*IPv6 over Low Power Wireless Personal Area Network*
ABAC	*Attribute-Based Access Control*
ACL	*Access Control List*
ACM	*Association for Computing Machinery*
AES	*Advanced Encryption Standard*
API	*Application Programming Interface*
CCS	*Computing Classification System*
CBC	*Cipher Block Chaining*
CFB	*Cipher Feedback*
CoAP	*Constrained Application Protocol*
CPU	*Central Processing Unit*
CSA	*Cloud Security Alliance*
CTR	*Counter Mode*
DES	*Data Encryption Standard*
DDoS	*Distributed Denial of Service*
DH	*Diffie-Hellman*
DoS	*Denial of Services*
DSA	*Digital Signature Algorithm*
DTLS	*Datagram Transport Layer Security*
E2E	*End-to-End*
ECB	*Electronic Code Book*
ECC	*Elliptic Curve Cryptography*
ECDH	*Elliptic Curve Diffie–Hellman*
ECDSA	*Elliptic Curve Digital Signature Algorithm*
GPS	*Global Positioning System*
HTTP	*Hypertext Transfer Protocol*
IEEE	*Institute of Electrical and Electronics Engineers*
ICMP	*Internet Control Message Protocol*
IdC	Internet das Coisas

IDS	Sistemas Detetores de Intrusões
IoE	*Internet of Everything*
IoT	*Internet of Things*
IP	*Internet Protocol*
ISM	*Industrial, Scientific and Medical*
jSSC	*Java Simpple Serial Conector*
LoRa	*Long Range*
LPWAN	*Low-Power Wide-Area Network*
M2M	Máquina-a-Máquina
MAC	Código de Autenticação de Mensagem
MD	*Message Digest*
MIT	*Massachusetts Institute of Technology*
MMU	*Memory Management Unit*
MQTT	*Telemetry Transport Message Queue*
OFB	*Output Feedback*
PaaS	*Platform-as-A-Service*
RAM	*Random Access Memory*
RC4	*Rivest Cipher 4*
RBAC	*Role-Based Access Control*
RFID	*Radio Frequency Identification*
REST	*Representational State Transfer*
RSA	*Rivest Shamir Adleman*
SASL	*Simple Authentication and Security Layer*
SDN	*Software Defined Networking*
SHA	*Secure Hash Algorithms*
SSL	*Secure Socket Layer*
TIC	Tecnologias da Informação e Comunicação
TCB	*Trusted Computer Base*
TCP	*Transmission Control Protocol*
TDES	*Triple Data Encryption Algorithm*
TI	Tecnologia da Informação

TLS	*Transport Layer Security*
UBI	Universidade da Beira Interior
UDP	*User Datagram Protocol*
URI	*Universal Resource Identifier*
VEA	*Vast Encryption Algorithm*
VPN	*Virtual Private Network*
WEP	*Wired Equivalent Privacy*
WoT	*Web of Things*
WPA	*Wi-Fi Protected Access*
WPAN	*Wireless Personal Area Networks*
WSN	*Wireless Sensor Network*
WWW	*World Wide Web*
XML	*eXtensible Markup Language*
XMPP	*eXtensible Messaging and Presence Protocol*

Capítulo 1

Introdução

Nas últimas décadas, a Internet, incluindo a sua infraestrutura de suporte e tecnologias, tem estado em constante evolução, sendo a *World Wide Web* (WWW) um sistema de documentos que permite a apresentação dos mesmos em forma de hipertexto, uma das facetas pela qual é mais conhecida. Com a evolução da Internet, este sistema tem evoluído gradualmente para a *Web 2.0*, o qual tem possibilitado a participação ativa e contribuição de utilizadores, empresas e sociedade de modo geral nesse mundo de informação. As tecnologia usadas neste sistema estão voltadas para a interação social moderna e para os negócios locais e globais (e.g., redes sociais, *blogs, e-commerce*, serviços de rastreio, serviços noticiosos, etc.). Paralelamente à rápida evolução e crescimento da Internet estão as tecnologias de redes de sensores, cujo aparecimento e viabilização tem fomentado e possibilitado a criação de novos conceitos, visões e, por sua vez, o desenvolvimento de novas tecnologias voltadas para interação entre os utilizadores e as máquinas. Os benefícios de automatizar os processos através da comunicação direta Máquina-a-Máquina (M2M) de dispositivos ligados à Internet permitiu que estes estivessem prontos e disponíveis para participarem na resolução de problemas diários afetos à sociedade. Esta visão tem alimentado o paradigma que chamamos de Internet das Coisas (IdC).

Embora não exista uma definição universal para a IdC, o conceito central é que os objetos do dia-a-dia sejam equipados com recursos de identificação, deteção, rede e processamento que permitam a comunicação entre si e com outros dispositivos e serviços pela Internet [Med16]. Os principais conceitos subjacentes à IdC não são novos. Durante anos, tecnologias como *Radio Frequency Identification* (RFID) e redes de sensores foram usadas em contextos industriais e de manufatura para rastrear vários itens. A ideia de comunicação direta M2M também não é nova, pois é básica para a ideia da Internet na qual clientes, servidores e *routers* comunicam entre si. O que a IdC representa é uma evolução do uso destas tecnologias existentes em termos de número e tipos de dispositivos, bem como a aproximação aos utilizadores finais e interconexão em redes destes dispositivos pela Internet. E.g., a maioria dos dispositivos atualmente ligados à Internet foram originalmente implementados para fazer parte da Internet e possuir recursos integrados de processamento, armazenamento e rede. Estes dispositivos incluíam servidores, *desktops, laptops, tablets e smartphones*.

O principal objetivo da IdC é permitir que dispositivos e sistemas do nosso quotidianos (e.g., recetores de áudio, televisões, eletrodomésticos, sistemas de videovigilância, sistemas de arrefecimento, sistemas de distribuição de energia, detetores de presença) estejam ligados a Internet para ganhos de funcionalidade e eficiência, mesmo que estes não tenham sido projetados inicialmente com esta capacidade em mente.

A IdC tem verificado um crescimento impressionante nos últimos anos, acarretando vários problemas. Estes problemas criam, portanto, vários desafios para a IdC. Um conjunto de restrições

dos próprios dispositivos alvo de estudo fazem com que estes desafios sejam de difícil resolução, bem como a diversidade de ambientes em que podemos conceber a sua instalação. Primeiro, os dispositivos IdC dificilmente passam por um controlo rígido relativamente ao aspeto físico, e isso aumenta as probabilidades de um atacante mal intencionado conseguir informações relevantes sobre os utilizadores; segundo, como a maior parte das comunicações é feita através de tecnologias de redes sem-fio, é possível que atacantes explorem as vulnerabilidades inerentes a estas tecnologias e que as usem contra os dispositivos da IdC. Por último, os dispositivos IdC são caracterizados por possuírem recursos limitados em termos de energia, memória e processamento, o que dificulta a implementação de mecanismos fortes de segurança.

Por outro lado, os serviços na IdC são baseados em processos de análise e recolha de dados que, pelas limitações dos próprios dispositivos, implicam normalmente que os dados são enviados através da rede para locais remotos às da coleção dos dados. Por si só, esta forma de atuar interfere na privacidade do utilizador. Esta forma de operação, nativa a muitos dos equipamentos da IdC, acumula pela negativa ao ecossistema existente da Internet, em que os utilizadores, em muitos os casos, precisam de divulgar ou compartilhar os seus dados pessoais para aceder a serviços ou sistemas na Internet.

Alcançar todos os requisitos de segurança e garantir a confiança na recolha, transmissão e processamento de dados da IdC é uma tarefa árdua e difícil de ser cumprida. O fornecimento inteligente de serviços sensíveis ao contexto e personalizados e, ao mesmo tempo, a preservação da privacidade, confidencialidade e integridade dos dados do utilizador a um nível esperado é visto como um grande desafio atual da IdC, quer em termos académicos, quer em termos industriais.

O escopo desta dissertação está sobretudo limitado à área da segurança informática, nomeadamente às sub-áreas de redes e segurança de sistemas, visando explorar o tema do estado da segurança na IdC, mais especificamente o desempenho de implementações existentes de algoritmos em plataformas populares deste paradigma à data da escrita da dissertação. Na versão de 2012 do ACM *Computing Classification System* (CCS), os tópicos que melhor descrevem esta dissertação podem ser definidos da seguinte forma:

- **Segurança e Privacidade – Segurança de Rede;**
- **Segurança e Privacidade – Segurança de software e aplicativos;**
- Segurança e Privacidade - Segurança de Sistemas;
- Redes - Serviços de rede.

1.1 Motivação

Nesta secção apresenta-se a motivação para conduzir o trabalho das perspetivas pessoal e técnica.

Do ponto de vista pessoal, o trabalho foi motivado pelo fascínio pelo mundo tecnológico, as abordagens que dele surgem inerentes ao desenvolvimento de tecnologias voltadas para a resolução de problemas reais, que alimentam a esperança de que o ser humano possa chegar a lugares que nenhum outro ser chegou. A forma como se aborda as questões que nos são importantes através das tecnologias varia de pessoa para pessoa. No entanto, através de estudos

dirigidos e a criação de mecanismos sustentáveis, vários problemas têm-se dissipado, resultando num mundo cada vez melhor. Após ter estudado um conjunto de tecnologias inovadoras durante o ano curricular e ter percebido que estas tecnologias apresentam um impacto profundo na sociedade de forma geral, envidou-se um esforço no sentido de contribuir com um trabalho que se concentrava na segurança em pelo menos numa destas tecnologias existentes atualmente. Entretanto, escolheu-se um paradigma que está em fase embrionária, mas que gradualmente vem apresentando avanços significativos, i.e., a Internet das Coisas. Neste sentido, foi impossível apresentar indiferença relativamente aos desafios inerentes a esta tecnologia. Portanto, além da vontade de aprender cada vez mais sobre este mundo tecnológico, foi mantida também a vontade de poder contribuir com este trabalho à medida em que esta tecnologia ganha destaque entre os utilizadores.

Do ponto de vista técnico e de investigação, deve dizer-se que as motivações para o trabalho apresentado se devem sobretudo à imaturidade da segurança no ecossistema IdC, à falta de conhecimento básico sobre como se aplicam mecanismos existentes em dispositivos potencialmente limitados computacionalmente e à falta de uma abordagem *bottom-up* e de verdadeira engenharia de segurança no desenvolvimento de sistemas informáticos ou software. Note-se que este trabalho constitui uma modesta contribuição ao vasto tema da segurança na IdC e foca-se na análise do estado da arte e no estudo prático da performance de mecanismos de segurança existentes numa plataforma muito popular da IdC. Motiva saber que este estudo poderá servir para melhor concluir acerca da adequação de mecanismos atuais aos novos dispositivos da IdC, bem como quais podem ser usados no imediato, para melhor compreender o problema em mãos.

1.2 Problema e Objetivos

Um dos problemas mais gerais que afetam a segurança de sistemas e software da IdC refere-se ao facto da engenharia de segurança não ser simplesmente feita para novos sistemas ou serviços. Este problema estende-se à área da informática em geral. De uma forma mais particular, outro problema que afeta a IdC é a falta de entendimento na forma como as limitações dos dispositivos podem inviabilizar a utilização de mecanismos já existentes para algumas plataformas utilizadas na concretização do paradigma. Dada a relevância do tema, foi proposto como principal objetivo deste trabalho identificar e estudar requisitos de segurança em IdC, e analisar a performance de alguns dos mecanismos que os preenchem numa plataforma reconhecidamente IdC, rumo ao estabelecimento de futuro mapeamento entre requisitos e mecanismos de segurança sob as restrições dos dispositivos da IdC.

A lista a seguir apresenta os objetivos principais de forma resumida e previstos no plano de trabalho:

- Enumerar os requisitos e mecanismos de segurança disponíveis no estado da arte;
- Testar o desempenho de tecnologias relacionadas à segurança em pelo menos uma plataforma de desenvolvimento para a IdC (e.g., *firewall*, sistema de detenção de intrusão e uma suite de criptografia);
- Começar o trabalho de mapeamento entre requisitos e mecanismos específicos de segurança sob as restrições dos dispositivos da IdC;
- Idealizar uma ferramenta para prova de conceito que integre este mapeamento.

1.3 Principais Contribuições

As contribuições deste trabalho constituem apenas uma parte do longo caminho a percorrer para uma IdC mais segura por desenho. Estas podem ser sucintamente, de forma redutora, resumidas da seguinte forma:

- É apresentado um esforço de identificação de requisitos de segurança para a área específica da IdC. Esta parte do trabalho encontra aplicação a montante do ciclo de desenvolvimento, acreditando-se ser necessário recomeçar várias vezes pelo estudo da engenharia da segurança para se conseguirem sistemas ou software seguro por construção;
- São discutidos e comparados trabalhos do estado da arte nesta área específica do conhecimento, de modo a perceber o seu estado atual de desenvolvimento e falhas;
- São apresentados um conjunto de testes de performance a algoritmos da criptografia moderna e a mecanismos muito utilizados em soluções de segurança informática, efetuados num dispositivo específico da IdC, um Raspberry Pi 3. Os resultados desta análise são comparados com os resultados obtidos num computador pessoal. Acredita-se que este trabalho dá uma ideia prática do problema inerente à transposição e aplicabilidade das tecnologias existentes em dispositivos da IdC;
- Por fim, é apresentado um mapeamento entre requisitos e tecnologias, bem como um esforço inicial para uma arquitetura de uma ferramenta para fazer esse mapeamento.

Algumas das contribuições mencionadas em cima serão o assunto de um artigo científico num futuro próximo.

1.4 Estrutura da Dissertação

Esta dissertação esta dividida em 5 capítulos que abordam sequencialmente os principais conteúdos endereçados no processo de investigação e desenvolvimento do projeto antes descrito. Cada um dos capítulos pode ser descrito da seguinte forma:

- O presente capítulo **(Capítulo 1)** aborda questões inicias de um documento com a configuração de uma dissertação, nomeadamente motivação, objetivos e as contribuições, fazendo o enquadramento do trabalho desenvolvido no âmbito do respetivo projeto de dissertação;
- O **Capítulo 2 – Estado da Arte, Definição e Caracterização da IdC** – apresenta conceitos fundamentais inerentes aos ambientes inteligentes do paradigma da IdC, desde a definição, desafios, e o conjunto de tecnologias envolvidas e que dão vida a este ecossistema. São também apresentados alguns estudos já realizados e relacionados com o tema em desenvolvimento nesta dissertação sendo que, no final, se procura fazer uma comparação entre os mesmos;
- O **Capítulo 3 – Requisitos e Mecanismos de Segurança em Internet das Coisas** – enumerada e descreve requisitos e mecanismos de segurança para a IdC, abordando também a necessidade de manter este ambiente seguro. É ainda apresentado neste capítulo um mapeamento entre estes requisitos e os vários mecanismos.

- O **Capítulo 4 – Testes em Plataformas da Internet das Coisas e Arquitetura para o Mapeamento** – apresenta e descreve com mais detalhe um conjunto de testes de performance feitos a uma série de algoritmos criptográficos encontrados na biblioteca *openssl*, realizados em dispositivos apropriados para a IdC (e.g., Raspberry Pi 3) e comparando os resultados obtidos com os obtidos num computador pessoal. Por fim, apresenta e descreve a arquitetura da ferramenta idealizada para prova de conceito integrando o mapeamento dos requisitos e mecanismos de segurança.
- O **Capítulo 5 – Conclusão e Trabalho Futuro** – apresenta as principais conclusões relativas ao trabalho que foi feito e, no final, apresenta de forma resumida um vislumbre sobre o trabalho futuro.

Capítulo 2

Estado da Arte, Definição e Caracterização da Internet das Coisas

2.1 Introdução

O paradigma IdC tem estado à ganhar muita popularidade nos últimos anos, sobretudo por causa da facilidade e mobilidade em ligar dispositivos portáteis a Internet e da partilha de informação que esta rede permite. Os utilizadores procuram realizar as suas tarefas de forma eficiente através de tecnologias que diretamente lidam com os vários problemas da sociedade, como é o caso da IdC. A IdC não só ajuda as pessoas realizarem eficientemente as suas tarefas, devido à imersão no próprio quotidiano, como também facilita o uso e a aprendizagem das novas tecnologias. Este capítulo apresenta, de forma abrangente, conceitos associados ao tema em desenvolvimento nesta dissertação, apresenta um pequeno resumo dos trabalhos já realizados e, por fim, compara-os com o intuito de apresentar os pontos chaves de cada trabalho. Assim sendo, a secção 2.2 apresenta uma abordagem geral sobre a IdC, descrevendo conceitos importantes envolvendo o tema. A secção 2.3 aborda o enquadramento dos trabalhos já existentes e apresenta uma análise comparativa entre eles. A secção 2.4 apresenta de forma resumida os riscos e ameaças emergentes na IdC.

2.2 Internet das Coisas, Visão geral

O termo IoT (em Português Internet das Coisas (IdC)) [Med16], foi introduzido no final da década de 90 por Kevin Ashton, um investigador britânico do *Massachusetts Institute of Technology* (MIT), que o usou para descrever um ambiente com objetos físicos ligados à Internet com objetivo de partilharem dados referentes ao estado da rede. A IdC é hoje considerada um ecossistema em que uma grande quantidade de objetos, sensores e dispositivos estão conectados através de uma infraestrutura de comunicação para fornecer serviços e recursos. O número de dispositivos conectados à Internet está a aumentar de forma acentuada também devido à evolução da IdC. Esses dispositivos incluem não só computadores pessoais, tablets, telefones inteligentes, e outros dispositivos de mão, como quase tudo que nos rodeia, desde que habilitados com capacidade de ligação à Internet e com poder de processamento de dados.

A maioria dos dispositivos móveis incorpora diferentes sensores que podem sentir, realizar, tomar decisões inteligentes e transmitir informações úteis através da Internet. A integração da IdC no quotidiano humano permite fácil identificação de problemas (por vezes automática), e podem contribuir na sua melhor e eficiente resolução. É claro o seu potencial de integração em todos os sectores da sociedade. No sector público, existem já aplicações para várias áreas, nomeadamente Saúde, Transportes, Educação, Agricultura e Serviços de Emergência. No sector privado, a IdC expande-se e afirma-se cada vez mais em processos industriais, permitindo a eficiência na produção assim como nas operações apresentadas pelas empresas, desde a inte-

ração dos funcionários com os clientes, como na prestação e inovação de serviços. A figura 2.1 apresenta, de forma simbólica e simplista, as aplicações, sectores ou áreas chaves da sociedade onde a IdC apresenta impactos positivos e significativos.

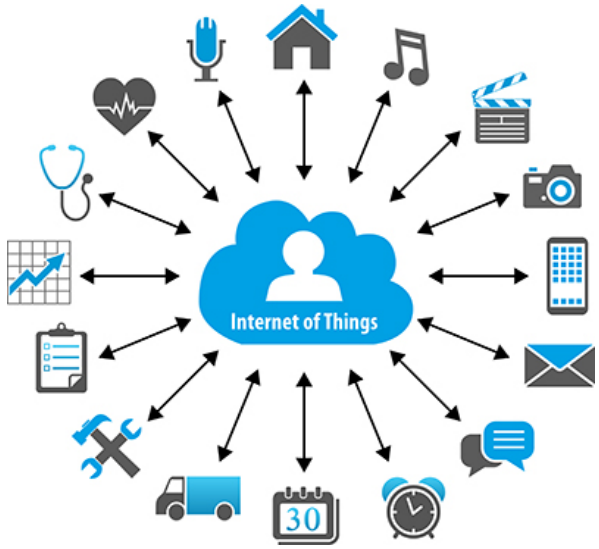

Figura 2.1: Internet das Coisas e suas utilidades nos vários sectores ou áreas chave [SV17].

Como anteriormente referido, a IdC tem melhorado o estilo de vida de muitas pessoas, sendo notável o avanço que tem, por sua vez, motivado no ramo das tecnologias de telecomunicações. No entanto, existe uma crescente preocupação no que diz respeito aos desafios inerentes ao ecossistema que lhe está subjacente, pois lida normalmente com informações sensíveis dos utilizadores. Um outro pormenor a acrescentar é que, geralmente, grande parte destes desafios surgem da necessidade de tornar a IdC um ambiente mais seguro tanto para os utilizadores como para os dispositivos a si ligados.

Muito tem sido feito para que a IdC se torne a próxima evolução da Internet, e assim permitir um maior aglomerado de dispositivos partilhando dados, serviços e recursos. Na base da evolução e do crescimento da IdC está a evolução das tecnologias de comunicação, e ainda os serviços de processamento e armazenamento prestados pela *cloud* por empresas do ramo, para além das empresas responsáveis pela fabricação de objetos inteligentes. A IdC faz uso de uma larga panóplia de tecnologias, algumas existentes há muitos anos, potencialmente lidando com uma grande quantidade de dados que se espalham por áreas geográficas de difícil delimitação. Visto de forma mais abrangente e levando em consideração as preocupações dos utilizadores relativamente aos seus dados (e.g., pessoais, bancários, saúde), pode-se afirmar que a segurança e a privacidade são desafios que se destacam entre os outros, por serem requisitos primordiais a se ter em conta em sistemas ligados a Internet exacerbados pelos fatores descritos antes.

Com o surgimento da IdC, vários outros conceitos começaram a ganhar interesse por parte dos utilizadores e a materializar-se no mundo da Internet, como é o caso do *Internet of Everything* (IoE) e do *Web of Things* (WoT). Por exemplo, o WoT funciona diretamente na camada de Aplicação da IdC, através do protocolo *Hypertext Transfer Protocol* (HTTP), que interliga vários dispositivos com o objetivo de aumentar a produtividade através das técnicas existentes na *web*, estendendo-as para cenários da IdC. Segundo Tein-Yaw Chung et al., [CMA+13] o WoT está cada vez mais a integrar objetos inteligentes com objetivo de disponibilizá-los como recursos

na *web* através de meios adequados. Assim como na IdC, a WoT deve sempre permitir e incluir as melhores práticas de segurança e privacidade, suportando os requisitos e mecanismos de segurança usados em sistemas da IdC.

2.2.1 Objetos Inteligentes e Tecnologia Máquina-a-Máquina

A IdC tem demonstrado ser uma tecnologia promissora e desempenhado um papel muito importante na resolução de vários problemas que afligem a sociedade. Na base da IdC está a tecnologia Máquina-a-Máquina (M2M), responsável por permitir e facilitar as comunicações entre dispositivos inteligentes sem a intervenção direta de mãos humanas [CWG+14]. Assim, o uso de dispositivos ligados à Internet através de um identificador (e.g., *Internet Protocol* (IP)) como é o caso de sensores, atuadores em habitações, empresas e indústrias, permitiu o crescimento de novos serviços, capazes de explorarem os problemas da sociedade e darem soluções concretas e efetivas. No entanto, tem havido uma ligeira confusão por parte dos utilizadores em relação aos termos IdC e M2M. O conceito de M2M está voltado especificamente para as comunicações entre dispositivos, geralmente através de comunicações ponto-a-ponto usando módulos de hardware específicos para redes com ou sem fios. Já o conceito da IdC está voltado para as informações que esses dispositivos geram, e como essas informações podem ser combinadas e utilizadas de acordo as necessidades. Tanto a IdC quanto a M2M realizam um papel preponderante na massificação não só da Internet como no desenvolvimento de novas tecnologias e na disponibilização de novos serviços.

Um outro termo importante neste contexto é o de *objeto inteligente*. Dia após dia, são adicionados vários objetos diferentes à IdC com finalidades diversas, dificultando a gestão de dados que são partilhados diariamente neste ecossistema. Quanto mais dispositivos estão ligados a uma rede, maior é a probabilidade de apresentar problemas de segurança. Os objetos ligados à IdC são considerados inteligentes por possuírem microprocessadores de grande capacidade, que reagem de modo programado a estímulos fazendo com que a resposta seja imediata ou próxima disso. Um objeto inteligente pode ser qualquer objeto físico (”Coisa”) que apresenta os seguintes requisitos: (i) ter uma identificação e possuir um endereço de rede (Internet); (ii) possuir alguma capacidade de receber e enviar informações para outros dispositivos; (iii) possuir poder de processamento em concordância com as suas actividades; e, por último, (iv) deve ter no mínimo um sensor para fenómenos físicos (e.g., calor). Pode afirmar-se que nem todos os objetos ligados a uma determinada rede são considerados inteligentes; mas sim apenas os que interagem fisicamente com a rede e que geram dados através dos sensores embutidos.

2.2.2 Desafios e Oportunidades da Internet das Coisas

Como discutido anteriormente, o rápido desenvolvimento e crescimento das tecnologias muito contribui para a massificação do que hoje é conhecido como IdC que, no entendimento de muitas pessoas, nos faz viver o futuro ainda no presente. No entanto, existe uma grande preocupação relativa à estabilidade e à regulamentação desta tecnologia. Dado o seu impacto e penetração na vida das pessoas e negócios, é de extrema importância que as empresas elaborem formas ou mecanismos para gerir os dados gerados de forma eficiente e segura. Enquanto que há um trabalho de consciencialização a fazer nos consumidores finais em relação aos produtos da IdC que compram, devem ser os fabricantes de sistemas e programadores de software para a IdC a procurar desenhar sistemas seguros por natureza. Os estudos já realizados no ramo, bem como a existência de possíveis soluções propostas por investigadores, não têm sido suficientes para

colmatar todos os desafios da IdC.

A questão da segurança e privacidade tem-se revelado a maior preocupação neste paradigma, tanto por parte de investigadores como utilizadores de forma geral. No entanto, saber até que ponto estes dispositivos são seguros, e com isso, encontrar maneiras de implementar mecanismos fortes de segurança, dará um avanço para estabilização da IdC. A quantidade de dispositivos e a quantidade elevada de dados partilhados pela IdC, permitem-nos perceber a dimensão desta questão, sob ainda a premissa de que o crescimento da IdC não está para abrandar nos próximos anos. De igual modo, as empresas de ramos tecnológicos afins, como a *Cisco, Gartner Group, IHS, e IDS*, argumentam que até ao ano de 2020, o número destes dispositivos excederá os 20 mil milhões, como mostra a figura 2.2.

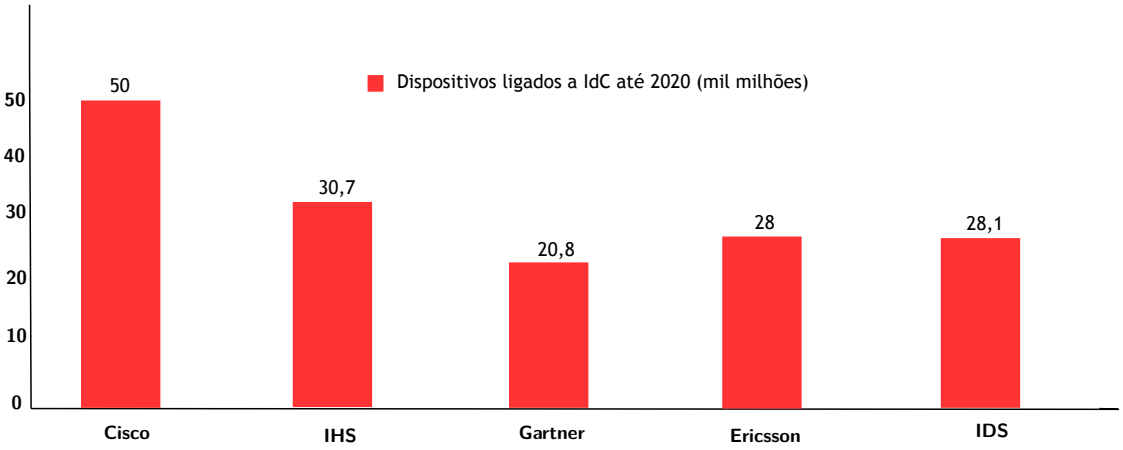

Figura 2.2: Crescimento previsto da IdC até ao ano de 2020 de acordo com [Nor16].

Vivemos numa década onde a coleção contínua de dados não se confina aos seres humanos. A massificação da ligação de objetos ativos ligados à rede gera um grande desafio para as empresas de IdC, em particular para as empresas fornecedoras de serviços e infraestruturas de rede. Como mencionado antes, um dos grandes desafios para a IdC é a segurança. Em baixo, e de forma resumida, são apresentados alguns desafios inerentes à IdC que aos poucos e com o avanço da tecnologia, e a realização de investigação aplicada, serão superados. Empresas como a *Cisco, Google, Microsoft*, etc., trabalham atualmente para solucionarem alguns destes desafios [Nal16]:

- ***Escalabilidade:*** a IdC tem a capacidade de operar numa escala maior de forma rápida e flexível quando comparada com as redes convencionais, através das tecnologias de comunicação específicas que utiliza, como apresentado na subsecção 2.2.5. No entanto, a maioria dos dispositivos inteligentes funcionam mais em ambientes locais e restritos [CM10]. Assim sendo, dada a quantidade e a variedade de dispositivos envolvidos em ambientes inteligentes e ligados à Internet, as aplicações voltadas para IdC devem ser desenvolvidas atendendo o requisito de escalabilidade e dinamismo esperado para estes ambientes. De outro modo, devem ser desenvolvidos mecanismos que permitam melhorar os serviços já existentes, e permitam ainda a expansão de novos serviços para ambientes inteligentes.

- ***Endereçamento:*** atualmente, os objetos inteligentes utilizam maioritariamente o protocolo de endereçamento IPv4 que é limitado a menos de 5 mil milhões de endereços. No entanto, com o aumento de computadores e outros dispositivos em redes convencionais, o número utilizável destes endereços diminuiu muito e, embora pouco se aborde o assunto, esta situação até certo ponto dificulta o progresso e fluidez da IdC. Prevê-se que os endereços IPv4 não serão suficientes para satisfazerem esta demanda de dispositivos interligados. Assim sendo, uma das soluções mais viável e que de modo geral resolveria este problema consiste na utilização generalizada do protocolo de endereçamento IPv6, que suporta uma melhor gestão dos endereços IPs e recursos de segurança aprimoradas. No entanto, encontrar maneira de tornar este protocolo o principal meio de endereçar dispositivos da IdC constitui ainda um desafio.

- ***Gestão de dados:*** um aspeto importante a ter em conta na IdC diz respeito à recolha e tratamento de dados. Diariamente e com bastante frequência, grandes quantidades de dados são gerados pelos dispositivos em ambientes inteligentes que criam uma sobrecarga e consequentemente, deixam estes ambientes vulneráveis [MO13]. Além disso, estas vulnerabilidades representam riscos para os negócios e para a sociedade em geral, caso não sejam resolvidas de forma apropriada, a partir de uma perspetiva de segurança, privacidade e bem-estar dos utilizadores. As soluções tecnológicas da IdC e os dados gerados pelos dispositivos, representam uma mudança significativa na maneira como estes são armazenados e posteriormente como são usados na tomada de decisões. A gestão de dados é cada vez mais uma prioridade para as empresas, já que os dados se tornam fundamentais para modelos de negócios em todos os setores;

- ***Segurança e privacidade:*** além de ser um requisito de segurança, é também um dos desafios mais problemáticos para a IdC, se não mesmo, o que necessita de maior atenção neste contexto. É imprescindível que entidades maliciosas não possuam acesso aos dispositivos em ambientes inteligentes, pois os prejuízos podem ser desastrosos para os envolvidos. No entanto, garantir segurança em ambientes que estão sempre a evoluir como é o caso da IdC é quase impossível, tudo porque não existem mecanismos fortes que lidam com as intermináveis ameaças e ataques direcionados a estes ambientes por causa do poder computacional reduzido que possuem. A segurança em ambientes inteligentes abrange vários aspetos importantes, nomeadamente a verificação de ameaças e riscos e posteriormente a utilização de mecanismos de segurança que inviabiliza estas ameaças. As soluções de segurança personalizadas oferecidas pela comunidade académica para a IdC são muitas vezes soluções pontuais o que, em termos teóricos, ajuda a entender o panorama geral da segurança da IdC, mas em termos práticos pouco contribui para a eficácia de mecanismos fortes de segurança para ambientes inteligentes [KT17]. No entanto, é inviável exigir que dispositivos da IdC com recursos restritos implementem mecanismos fortes em toda a sua extensão. Embora um único mecanismo de segurança não atenda a todas as necessidades da IdC relativas a segurança, é possível adaptar os mecanismos existentes para suprirem algumas falhas encontras neste ecossistema;

- ***Fonte de Energia:*** dispositivos e sensores ligados à IdC necessitam de fontes de energia para realizarem as suas funções, e poucos são ainda os dispositivos autossustentáveis, o que constitui um problema, porque na sua maioria as fontes de energia são baterias. A limitação no tempo de duração da carga constitui um dos outros problemas. A gestão de energia é um tópico bastante amplo e abrangente no contexto da IdC, sendo que alguns

dispositivos possuem restrições na coleta e consumo de energia; já outros parecem possuir maior adaptabilidade técnica de geração de energia, ou seja, usam a coleta de energia não só para se tornarem operáveis como também para aumentar ou estender a vida útil da bateria. Contudo, é de se esperar que a IdC dê um passo em frente neste ponto, permitindo que novas tecnologias sejam desenvolvidas com o objetivo de serem resolvidos estes problemas da limitação de carga apresentados pelos dispositivos inteligentes;

- ***Computação na nuvem:*** a IdC e o conceito da computação na nuvem andam normalmente de mãos dadas. Esta integração permite que os dispositivos inteligentes façam *offloading* de tarefas de processamento e tenham um melhor funcionamento e utilizem vários serviços e recursos de forma cómoda e segura [RSS+12]. Se usada corretamente, a utilização de serviços na nuvem traz vários benefícios para os utilizadores, desde a eficiência de energia até a otimização de recursos de hardware e software. Em geral, o uso destes serviços e recursos na nuvem equilibram restrições computacionais dos objetos inteligentes (e.g., em termos de armazenamento, processamento, energia, comunicação). Por outro lado, estes benefícios estendem-se também para as empresas prestadoras de serviços na nuvem. à medida que a IdC vai aderindo estes serviços, as empresas vão estendendo os seus objetivos para lidarem com os objetos inteligentes do mundo real de forma mais distribuída e dinâmica, e ainda para fornecerem serviços cada vez mais inovadores num grande número de cenários da vida real;

- ***Interoperabilidade:*** a falta de padronização nas comunicações entre os dispositivos da IdC tem contribuído para fraca massificação deste ecossistema. Dado haver um crescimento notável destes dispositivos, muitos protocolos e padrões foram desenvolvidos, gerando um problema de interoperabilidade que pode ser atribuído principalmente à existência de um conjunto diversificado de tecnologias antigas e novas. A interoperabilidade é fundamental na IdC, já que a maior parte das comunicação é feita entre dispositivos. Acrescentando a esta dificuldade, estão as restrições não compatíveis com normas que as empresas implementam nos dispositivos, limitando o seu raio de ação, permitindo por outro lado que os dispositivos inteligentes apresentem suas particularidades e habilidades, como a comunicação e a produção de dados [Hua16]. Contudo, estes dispositivos devem manter a comunicação e a cooperação independentemente das restrições que apresentam, portanto, há a necessidade de padronizar tecnologias de comunicação e processamento da IdC. Isto permitirá maximizar a inovação deste domínio, da mesma forma que a Internet o fez para os serviços de informação e comunicação.

2.2.3 Arquitetura da Internet das Coisas

A arquitetura da IdC tem de ser tal que permita lidar com grandes quantidades de dados, trocadas diariamente pelos dispositivos de comunicação de curto alcance, sensores e atuadores existentes. A arquitetura básica da IdC é normalmente dividida em três camadas, nomeadamente aplicação, rede e perceção [AOHA17]. No entanto, as três camadas usadas para definir a arquitetura básica são bastante gerais, e em muitos aspetos gera discórdias entre os investigadores da área. Com o intuito de aprofundar a visão geral da arquitetura da IdC, foi proposto em [WLL+10] uma arquitetura com cinco camadas: camada de negócio, camada de aplicação, camada de gestão de serviços, camada de rede e camada de perceção, como mostra a figura 2.3.

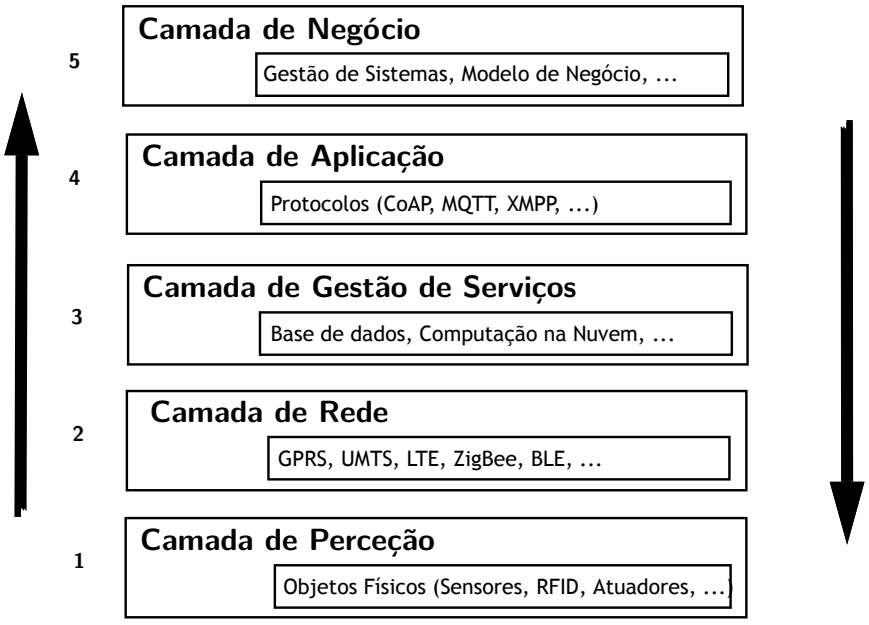

Figura 2.3: Arquitetura em camadas da IdC

Estas cinco camadas podem ser descritas sumariamente da seguinte maneira:

1. *Camada de Perceção:* considerada como a camada mais baixa da arquitetura da IdC, a sua finalidade é perceber os dados envolvidos no ambiente através dos dispositivos interligados, ou seja, é a que permite detetar, colecionar e processar os dados do ambiente, transformá-los em sinais digitais e enviá-los para a camada superior (e.g., camada de rede). Estes dados podem ser sobre o ambiente da rede (e.g., segurança, temperatura, humidade, localização), assim como um outro dado relevante para o contexto em que o objeto está contido [WLL+10]. É nesta camada que são encontrados dispositivos como RFID, sensores, atuadores ou etiquetas de código de barras;

2. *Camada de Rede:* é responsável por receber as informações vindas da camada anterior e determinar a rota pela qual estes dados chegarão até a camada superior através das redes integradas [LYZ+17]. Esta camada é geralmente conhecida como a parte mais consistente da arquitetura da IdC por permitir a transmissão de informações a longas distâncias, e por processar de forma inteligente as informações massivas. Nesta camada funcionam tecnologias como 3,4/G, Wi-Fi, ZigBee, Bluetooth LE, ...;

3. *Camada de Gestão de Serviços*: é responsável pela prestação de serviços referentes as informações provenientes da camada inferior (camada de rede). Esta camada identifica o pedido de um utilizador e fornece o serviço específico através dos protocolos de rede. Por outro lado, esta camada extrai informações da enorme quantidade de dados obtidos pelos dispositivos para produzir um resultado mais coerente e significativo. Além disso, esta camada é responsável pela mineração de dados e inclui normalmente análise, computação na nuvem e computação ubíqua [WLL+10];

4. *Camada de Aplicação*: é responsável pela união das tecnologias da IdC referentes aos protocolos de comunicação e aquelas especificamente direcionadas para os vários setores da sociedade. A importância desta camada para IdC é que ela tem a capacidade de

facilitar o acesso aos serviços inteligentes para atender as necessidades dos utilizadores. Nesta camada, as aplicações implementadas pela IdC podem estar dirigidas para saúde, agricultura, residências, cidades, transportes, etc. [KKZK12];

5. *Camada de Negócio*: é responsável por construir um modelo de negócio e pela monitorização através, e.g., da criação de gráficos com base nos dados provenientes das camadas subsequentes. Além de projetar, analisar, implementar, avaliar, monitorizar e desenvolver elementos do sistema IdC, esta camada possibilita o suporte a processos de tomada de decisão com base na análise dos dados obtidos [WLL+10];

2.2.4 Aplicações da Internet das Coisas

Atualmente, a IdC vem melhorando os serviços de utilidade pública, trazendo vários benefícios nos distintos setores da sociedade, permitindo que milhões de pessoas usufruam de forma direta ou indireta destes benefícios. Para usufruir destes benefícios, não basta apenas que os dispositivo estejam interligados, ou seja, não basta ter uma câmara inteligente, um carro inteligente, uma residência lotada de dispositivos inteligentes, é necessário que, em função da necessidade do consumidor, estes dispositivos estejam ligados a um *gateway* e este à Internet. Debate-se largamente a questão sobre até que ponto estes dispositivos garantem a segurança de e para os utilizadores. No entanto, as questões relacionadas com a segurança não têm parado totalmente a evolução desta tecnologia. A seguir, são apresentados alguns benefícios tangíveis que a IdC proporciona ou pode vir a proporcionar nos setores chaves da sociedade:

- *Saúde:* constitui um dos setores mais importantes da sociedade, e um dos elementos vitais para o ser humano. A IdC na saúde tem estado a melhorar significativamente a qualidade dos serviços, a vida dos pacientes e a permitir eficiência nos diagnósticos de várias doenças. Como exemplo, através de sensores específicos, os pacientes não críticos podem ser monitorizados em casa sem a intervenção direta dos médicos, reduzindo a pressão sobre os recursos dos hospitais (e.g., médicos, recursos físicos). Ainda, a IdC voltada para a saúde melhora o acesso aos cuidados de saúde para pessoas que estão muito afastadas dos centros urbanos. As aplicações inteligentes voltadas para saúde permitem ainda aos pacientes possuírem melhor controlo sobre o seu estado de saúde em todos os momentos e em tempo real [LL16];

- *Residências:* as residências inteligentes têm potencial de impacto direto e específico na vida dos utilizadores. Estes ambientes recebem o adjetivo de *inteligente* porque interligam muitos sistemas domésticos (e.g., iluminação, temperatura, eletrodomésticos, segurança, equipamentos multimédia) e estes são controlados e monitorizados através de uma interface/sistema apropriado (e.g., telefone, computador, *tablet*), independentemente da hora e local [LG17]. Portanto, ambientes domésticos inteligentes proporcionam segurança para os residentes, conforto, eficiência, assim como economia no consumo de energia, água, etc.;

- *Cidades:* as cidades inteligentes são conhecidas por utilizarem as tecnologias em prol do desenvolvimento sustentável da sociedade em geral. Fazem o uso consciente das Tecnologias da Informação e Comunicaçãos (TICs), envolvendo a azáfama e pessoas da cidade na tomada de decisões. Uma cidade inteligente está equipada com diferentes elementos eletrónicos distribuídas por várias lugares, o que constitui um dos objetivos principais deste

setor, como sistemas de vídeo vigilância, sensores para sistemas de transporte, parques de estacionamento inteligentes, sistemas inteligentes de temperatura, etc.;

- *Transporte:* A IdC é considerada uma peça essencial para as cidades inteligentes, pois elas permitem a criação de soluções inovadoras, através de tecnologias avançadas de comunicação e sistemas de gestão de transportes em tempo real para circulação de veículos, pessoas e animais. O objetivo, além da eficiência na circulação de veículos e pessoas, é também garantir a segurança do tráfego, facilitar a resolução dos problemas relacionado ao trânsito, melhorar a qualidade do meio ambiente e melhorar ainda a taxa de utilização de energia;

- *Agricultura:* a agricultura é uma fonte de rendimento e de sustento de muitas pessoas em diferentes partes do mundo. Os recentes desenvolvimentos das TICs e das redes de sensor sem fio tornaram a manutenção e o funcionamento de indústrias agro-baseadas como estufas, floricultura e horticultura, mais fáceis. Com a implementação da IdC na agricultura, espera-se que os avanços neste setor sejam significativos, permitindo maior controlo na distribuição e utilização de fertilizantes e pesticidas, assim como no controlo do clima ambiental e produção de hortícolas.

2.2.5 Protocolos e Tecnologias de Comunicação

A visão da IdC é usar tecnologias inteligentes como a M2M para conectar objetos de alguma forma e em qualquer lugar. Assim sendo, é importante conhecer como esses dispositivos comunicam-se entre si e trocam informações, que para os utilizadores podem ser de caráter importante. Estamos num mundo onde cada vez mais as informações são tratadas como parte de negócios. Esta secção aborda as principais tecnologias de comunicação utilizadas em dispositivos IdC, identificando as características mais relevantes de cada um deles e a sua importância na infraestrutura de rede.

Na IdC, existe uma quantidade variada de tecnologias de comunicação. A comunicação entre dispositivos IdC depende de tecnologias que operam a vários níveis: tecnologia usada para conectar objetos e dispositivos a redes de comunicação, tecnologias com capacidade de interagir com os objetos e delas detetar mudanças no seu estado físico e tecnologias que permitem de forma rápida detetar os acessos por utilizadores não autorizados. Por tanto, é de extrema importância conhecer algumas tecnologias quando se estuda segurança na IdC. Em baixo são abordadas tecnologias de comunicação com mais detalhe:

- *Wi-Fi:* uma tecnologia de comunicação sem fios muito usada no mundo inteiro que está presente em escolas, casas, locais de trabalho e espaços públicos e comerciais. É muito utilizada em dispositivos IdC. O padrão que acompanha essa tecnologia é o *Institute of Electrical and Electronics Engineers* (IEEE) 802.11 com versões como IEEE 802.11a, IEEE 802.11b, IEEE 802.11n a operar nas frequências de 2.4/5 GHz, e o 802.11ac, que opera na frequência dos 5 GHz. A maioria desses padrões suportam taxas de transferências de dados de 1 Mb/s a 1 Gb/s [SKL17]. O alcance desta tecnologia sem fios varia entre 20 a 150 metros. Sendo desenvolvido como uma alternativa ao padrão Ethernet, espera-se que muitos dispositivos adotem esta tecnologia como o principal protocolo de comunicações na IdC.

- *ZigBee:* uma tecnologia regida pelo padrão IEEE 802.15.4 ainda recente, capaz de interligar mais de 150 dispositivos numa única rede [KM17]. A tecnologia ZigBee opera na frequência 2.4 GHz, mas pode operar também em frequências como 868 MHz e 915 MHz, alcançando taxas de transferência de 250 kbps. A ZigBee está a massificar-se e sua utilização na IdC depende normalmente de um *gateway* responsável por permitir a comunicação entre os dispositivos que usam a tecnologia e os que não.

- *MQTT:* acrónimo de *Telemetry Transport Message Queue* (MQTT), é um protocolo de mensagens que funciona na camada de aplicação da arquitetura IdC muito utilizado atualmente em dispositivos com tecnologia M2M e com requisitos de processamento reduzidos. Além de ser um protocolo padrão para a IdC, é também considerado como um protocolo de rede leve, flexível e ideal para as aplicações da IdC. O MQTT funciona no modelo de publicação e assinatura *(publish/subscriber)*, que basicamente define dois tipos de entidades na rede, um intermediário central, designado normalmente de *Broker* ou servidor, e vários clientes ou assinantes. O *Broker* tem a missão de filtrar as mensagens enviadas pela rede com base num tópico especifico e redistribui-la aos outros assinantes. Em termos de segurança, o MQTT suporta vários mecanismos de autenticação e segurança de dados e da parte dos clientes essa responsabilidade é atribuída a cada cliente, ou seja, cada cliente responsabiliza-se pela segurança do seu dispositivo [YSAAH17]. A figura 2.4 ilustra o funcionamento básico do MQTT usando o modelo *publisher/subscriber*.

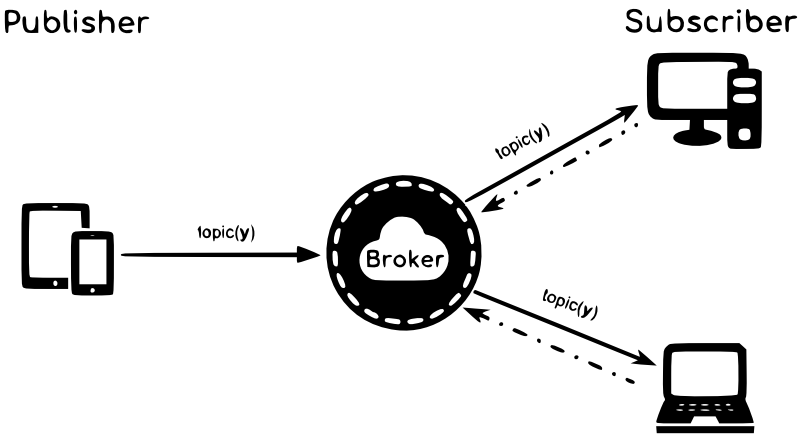

Figura 2.4: Funcionamento básico do MQTT, modelo *(publisher/subscriber)*.

- **CoAP:** o *Constrained Application Protocol* (CoAP) é um protocolo de rede que funciona na camada de aplicação da arquitetura da IdC para dispositivos com capacidades de processamento reduzidas. É por vezes referido como uma nova abordagem para protocolos de aplicativos da Internet que utilizam recursos muito limitados. Este protocolo foi desenvolvido com o intuito de funcionar junto com o HTTP e serviços *web Representational State Transfer* (REST). Entretanto, enquanto que o MQTT faz o uso de tópicos para o encaminhamento de mensagens, o CoAP faz uso do *Universal Resource Identifier* (URI) [TMV+14]. O REST é um dos elementos indispensáveis para o protocolo, permitindo aos clientes e servidores utilizarem os serviços web de forma simplificada e eficiente através dos URIs e métodos HTTP *(put, post, get, delete)* para indicar ações. O CoAP utiliza o *User Datagram Protocol* (UDP) e fornece mecanismos de segurança baseados no *Datagram Transport Layer Security* (DTLS). A Figura 2.5 ilustra o funcionamento básico do CoAP.

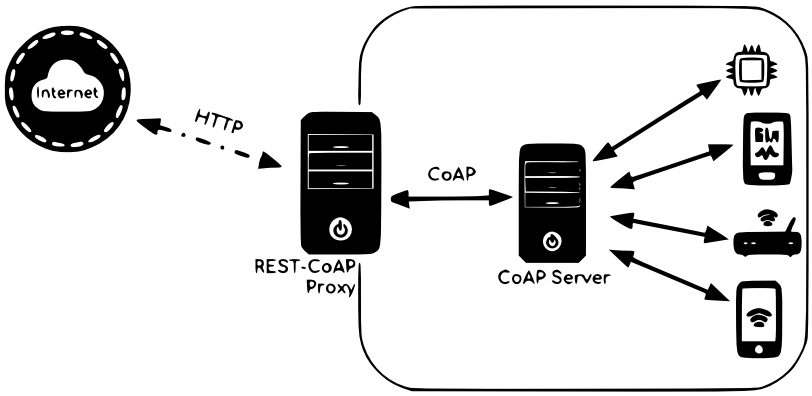

Figura 2.5: Funcionamento básico do CoAP.

- *LoRaWAN:* É um protocolo que funciona na camada lógica de rede baseando-se na tecnologia *Long Range* (LoRa), e oferece suporte a comunicação de longo alcance a vários dispositivos inteligentes possibilitando novos tipos de serviços em ambientes IdC. Tecnologias como LoRaWAN e Sigfox combinam baixo consumo de energia com longo alcance (ver tabela de comparação 2.1) e são chamados de *Low-Power Wide-Area Network* (LPWAN). O desenvolvimento de sistemas para soluções IdC para ambientes mais amplos, como casas inteligentes e cidades inteligentes é um campo em que o LoRa, junto com o Sigfox, têm um papel especial. A LoRa é uma tecnologia sem fios direcionada especificamente para a IdC que permite a comunicação entre dispositivos inteligentes a distâncias consideráveis (e.g., em áreas urbanas tem 2-5 Km de alcance, e em áreas rurais, até 15 Km); a banda de comunicações utilizada pertence à banda de frequência *Industrial, Scientific and Medical* (ISM) não licenciada. O LoRa usa uma topologia de rede em estrela, onde cada dispositivo pode comunicar diretamente com o módulo *Gateway*. A figura 2.6 apresenta a arquitetura básica do LoRaWAN.

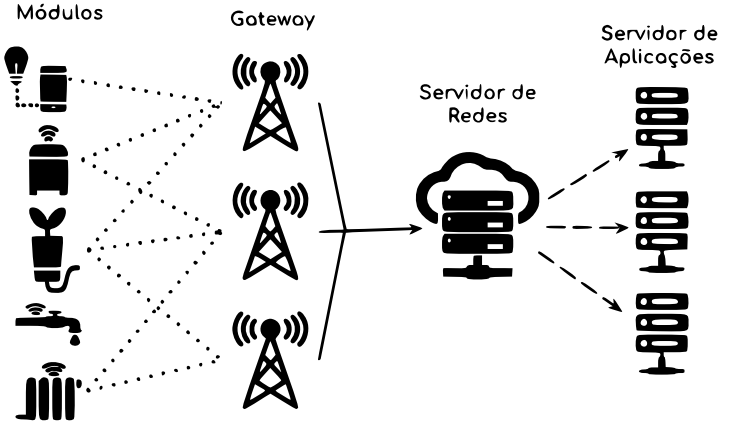

Figura 2.6: Arquitetura do protocolo LoRaWAN.

O protocolo LoRaWAN é uma pilha de rede padrão aberta e baseada utilização da LoRa na camada física. As especificações do LoRaWAN definem três tipos de dispositivos divididas em três classes (*A*, *B*, e *C*). A classe *A* e a classe *B* são praticamente alimentadas por baterias, enquanto que a Classe *C* é alimentada pela rede elétrica; *A* tem capacidade limitada para receber mensagens. Um nó só pode receber uma mensagem de *downlink* dentro de um intervalo de tempo de receção. Há duas janelas de receção entre 1 e 15 segundos após o envio de uma mensagem e outra, uma segunda duração após a primeira

janela de receção. A principal diferença nas três classes está na receção de pacotes (ou seja, enviados do gateway para o dispositivo). Os equipamentos LoRa funcionam com ondas de rádio de baixa frequência. Na Europa, operam nas bandas 867-869 MHz e utilizam uma transferência de dados que vão desde os 0.3 kbps aos 50 kbps [NOSALS18].

- *XMPP:* o *eXtensible Messaging and Presence Protocol* (XMPP) é um protocolo que permite a troca de dados extensíveis e estruturados em tempo real entre dois ou mais nós na rede [SA11]. Este protocolo é composto por um conjunto de tecnologias abertas usadas principalmente para troca de mensagens instantâneas, chamadas de vídeo e voz entre outras, já que suporta mensagens pequenas e de baixa latência. Estas características fazem do protocolo uma boa opção para comunicações em ambientes IdC. A comunicação entre os vários dispositivos com a base cliente-servidor são trocadas através do *eXtensible Markup Language* (XML). A sua arquitetura é a de cliente-servidor (esquematizada, de forma genérica, na figura 2.7), onde os clientes solicitam conexão ao servidor para interagirem com outros clientes, através da troca de mensagens. É comum combinar o XMPP com o protocolo *Transport Layer Security* (TLS) e *Simple Authentication and Security Layer* (SASL) (que serve como um mecanismo de autenticação) para garantias relacionadas com segurança.

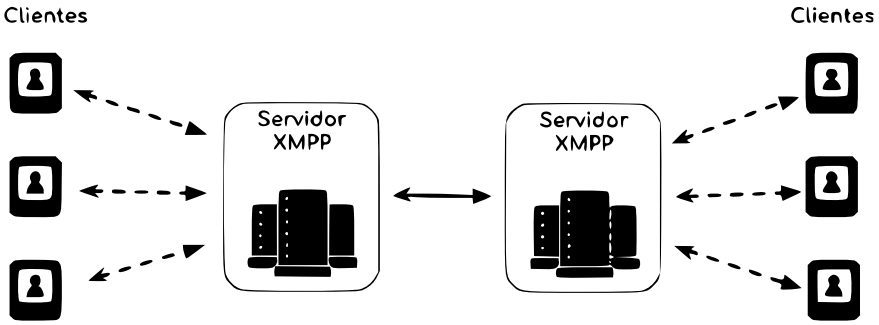

Figura 2.7: Arquitetura do protocolo XMPP.

- *Z-Wave:* É um protocolo de comunicação sem fios de baixa potência, mais usado para controlo remoto em ambientes residenciais e comerciais. Pensado para transmissão de pequenos pacotes de dados com uma taxa de transferência de 100 kps, um alcance de 30 m, e com uma topologia em malha. É Portanto adequado para comunicação entre dispositivos IdC.

- *Bluetooth LE:* Também conhecido como o padrão IEEE 802.15.1, é uma tecnologia que surge para a diminuição efetiva do consumo de energia dos dispositivos que suportam essa tecnologia. Funciona numa banda de frequência de 2.4 GHz, permitindo os utilizadores controlarem os objetos a partir dos seus dispositivos móveis.

- *RFID:* É uma tecnologia que usa radiofrequência para identificar e capturar dados de objetos. Pode ser usada em aplicações de redes residenciais, mas no entanto é muito usado em ambientes comerciais. A tecnologia RFID possui três bandas de frequências, a baixa frequência de 125 kHz, a alta frequência de 13 MHz, e a ultra frequência, que opera nas faixas de 860 MHz a 960 MHz respetivamente. O alcance desta tecnologia pode variar de 10 a 200 m, fornecendo ainda uma taxa de dados de até 4 Mbps [KPR16].

- *6LoWPAN:* A tecnologia *IPv6 over Low Power Wireless Personal Area Network* (6LoWPAN), é um protocolo que permite a comunicação e a troca de dados entre dispositivos de baixo consumo de energia através de uma rede já existente como uma *Wireless Personal Area*

Networks (WPAN). A tecnologia 6LoWPAN funciona numa banda de frequência de 2,4 GHz, 915 MHz (América do Norte) e 868 MHz (Europa), a transferência de dados vão desde os 20 Kbps, 40 kbps e 250 kbps, tendo um alcance de 10 a 100 m [RPK11].

A tabela 2.1 apresenta, de forma resumida e estruturada, uma comparação entre as várias tecnologias de comunicação antes descritas (os protocolos não são incluídos na tabela).

Tabela 2.1: Comparação entre as tecnologias de comunicação para a IdC abordadas. A tabela não se reporta aos protocolos.

	Parâmetros						
Tecnologia	Proprietário	IEEE	Alcance	Frequência	Transferência	Segurança	Topologia
Wi-Fi		802.11	150 m	2.4 GHz, 5 GHz	1 Gbps		Estrela
ZigBee		802.15.4	100 m	2.4 GHz	250 Kbps		Estrela
Z-Wave		✗	30 m	908.47 MHz	100 KB		Malha
Bluetooth LE		802.15.1	10/200 m	2.4 GHz	1 Mbps		Estrela
6LoWPAN		802.15.4	10/100 m	2.4 GHz	250 Kbps		Estrela
MQTT		802.15.4	100 m	2.4 GHz	250 Kbps		Estrela
CoAP		802.15.4	100 m	2.4 GHz	250 Kbps		Estrela
LoRaWAN	LoRa Alliance		2-15 km	870 MHz	0.3-50 kbps	AES-128 E2E	Estrela
NFC		802.15.4		2.4 GHz			Estrela

2.3 Análise de Segurança e Privacidade

Atualmente não existe um sistema ligado à Internet que garanta total segurança, e os sistemas IdC herdaram este problema. A IdC liga muitos recursos pessoais ou objetos de potencial valor elevado, o que traz grandes responsabilidades e riscos significativos para privacidade e segurança. Estas áreas representam desafios relevantes para a implantação de mecanismos fortes e seguros. A área da IdC é uma área em desenvolvimento e em plena expansão, afirmando-se como um paradigma promissor. Contudo, também pelo facto de ser recente, ainda existem poucos estudos nos campos específicos que esta dissertação foca. Esta secção descreve vários trabalhos disponíveis na literatura da especialidade, não sendo sobremaneira exaustiva. Vários artigos foram descartados por causa da relevância e da consistência na abordagem de temas que não davam muita informação sobre segurança e que estavam relacionados com a IdC. Assim sendo, a seguir, são apresentados várias estudos e trabalhos relacionados ao tema e aos objetivos desta dissertação, sendo feita uma comparação entre os trabalhos no final. Os riscos e ameaças que afligem diariamente tanto este ecossistema como os utilizadores são discutidos brevemente após essa comparação.

2.3.1 Trabalhos Relacionados

A literatura da especialidade tem proliferado na interseção das áreas da IdC e segurança recentemente. Assim, esta subsecção foca-se apenas na descrição de trabalhos relacionados e muito recentes (posteriores a 2014), comparando-os no fim.

Alex Akinbi et al. [AP15] apresentam um estudo sobre o mapeamento dos requisitos de segurança para identificar as áreas críticas nas camadas do modelo de desenvolvimento na nuvem *Platform-as-A-Service* (PaaS), mais focado no desenvolvimento de uma abordagem de gestão de riscos através destes requisitos. A abordagem utilizada realça as categorias para segurança deste modelo fornecidas pela *Cloud Security Alliance* (CSA)[1]. Os autores atribuem designações a estas categorias com o objetivo de facilitar a encontrar a área com o maior défice de segurança, e através de uma fórmula matemática, usando os valores numéricos, os autores criam um cenário numa empresa real, encontrando a área com maior preocupação no seu sistema de informação. Assim, os autores concluem que em função do estudo apresentado, os clientes em nuvem PaaS devem entender e identificar as áreas que necessitam de maior atenção em função dos requisitos de segurança estabelecidos.

Em [ATANM+16], Majid A. Al-Tae et al. fazem um estudo sobre o mapeamento dos requisitos de segurança no sistema móvel de saúde *mHealth* apresentado em [ATANMAA16]. O sistema permite que os profissionais de saúde e os pacientes monitorizem, através de sensores, a autogestão de glicose no sangue do paciente. Assim como analisado por Alex Akinbi et al., em [AP15], o estudo apresentado por Majid A. Al-Tae et al. baseia-se nos requisitos de segurança do modelo para nuvem PaaS, procurando encontrar a área crítica através dos mecanismos existentes nesse modelo e procura mostrar a necessidade de se adotar os mecanismos de segurança na implementação de sistemas seguros da IdC para a saúde, dando maior ênfase às propriedades de segurança.

Em [ADAS17], Waqar Ali et al. fazem o estudo das variadas ameaças e ataques de segurança num ambiente doméstico inteligente e avaliam o impacto negativo causado por estas ameaças relativamente à segurança num âmbito geral do sistema. Identificam ainda os requisitos e

[1] Organização que promove o uso das melhores práticas para garantir a segurança na *Cloud Computing*.

soluções de segurança no ambiente doméstico inteligente e com isso fazem um mapeamento entre os requisitos de segurança para IdC e as ameaças que violam esses requisitos baseando-se em vários cenários criados.

Em [LYZ+17], os autores Jie Lin et al. abordam de forma sucinta conceitos relacionados com a IdC, desde o impacto que esta tecnologia traz para sociedade, assim como aspetos relacionados com a segurança dos dispositivos habilitados para esta tecnologia. O artigo ainda aborda a arquitetura em camadas da IdC (e.g., perceção, rede e aplicação), nelas enquadrando o estudo. São abordados os desafios de segurança que os autores consideram como sendo os principais para cada camada. O artigo apresenta também medidas de segurança como forma de reduzir ou mesmo extinguir a facilidade que as entidades maliciosas possuem para se apoderarem estes sistemas; medidas estas que passam pelos requisitos de segurança e implementação de mecanismos de segurança para esta tecnologia. O mesmo estudo é apresentado em [DKS17] por M. Daud et al. que, de forma abrangente, usa estes conceitos para apresentar as vulnerabilidades existentes em duas das tecnologias fixas ligadas a ambientes da IdC como RFID e *Wireless Sensor Network* (WSN). Além disso, os autores interligam as ameaças à falta de consistência dos mecanismos de segurança, causando impactos de caráter negativo do ponto de vista da proteção dos dados, da privacidade do utilizador e da desestabilização da economia envolvente. É proposto, no final, um modelo de segurança baseado nas propriedades de segurança conhecidas como a tríade da segurança da informação, e estendidas noutros requisitos para suportarem toda a estrutura da IdC garantindo maior confiança, privacidade e permitindo uma comunicação segura entre os intervenientes.

Em [JVW+14] os autores analisam as questões de segurança ligadas à IdC, e abordam detalhadamente as principais tecnologias e os problemas de segurança existentes nas camadas da sua arquitetura, propondo como solução uma arquitetura fiável para minimizar os riscos de segurança a que estes ambientes estão sujeitos e, de uma maneira generalista, comparam esses problemas com os problemas de segurança de uma rede tradicional. F. Ye e Y. Qian apresentam, em [YQ17], apresentam uma proposta de uma arquitetura de segurança voltada para a IdC e aplicada em vários cenários, dividida num módulo de auditoria e dois de gestão de segurança, sendo o primeiro baseado em *Software Defined Networking* (SDN) e o segundo baseado num mecanismo de criptografia. A abordagem de segurança aplicada nessa arquitetura permite que os dispositivos inteligentes conectados à rede sejam separados logicamente da Internet, evitando que as vulnerabilidades a nível de hardware sejam exploradas remotamente. Portanto, os autores garantem que esta proposta aumenta a segurança dos ambientes IdC, desde as aplicações de interface ou gestão até aos dispositivos conectados à rede.

Os autores S. Oh, e Y. Kim [OK17] focaram a sua investigação em torno da segurança da IdC e os requisitos comummente usados como propriedades indispensáveis na elaboração e implantação de mecanismos de segurança fortes neste ecossistema. Dispositivos inteligentes conectados a ambientes IdC apresentam geralmente várias vulnerabilidades de segurança e funcionamento, aumentando a superfície de ataque, como de resto é abordado no trabalho de M.Daud, Q. Khan e Y. Saleem em [DKS17]. Estes autores propõem alguns dos requisitos básicos e fundamentais de segurança baseando-se em três características intrínsecas da IdC (heterogeneidade, redução da capacidade de processamento e no seu ecossistema dinâmico) e em seis elementos fundamentais para o alavancar desta tecnologia tanto nos dias de hoje como no futuro próximo (rede, nuvem, utilizador, ataques, serviços e plataforma). Em [Kim15], J. Kim aborda os requisitos de segurança

em aplicações da IdC sob a perspetiva do sistema *gateway*, para ele fundamental para solucionar as vulnerabilidades existentes em dispositivos inteligentes.

Em [HFH15, MPY17], os autores apresentam detalhadamente diversas análises sobre a segurança em ambientes IdC baseando-se nos problemas, vulnerabilidades, ataques e métodos de ataques recorrentes neste paradigma, exploram as limitações existentes nos dispositivos IdC e, as atuais questões de segurança ligadas a privacidade em ambientes IdC, e discutem ainda as principais preocupações de segurança deste ecossistema focando-se de forma mais exaustiva nos requisitos de segurança. Os autores deixam ainda claro que analisar e entender as necessidades de segurança que os ambientes IdC apresentam ajuda na implementação de mecanismos eficientes, fortes e seguros.

Em [HCLY16], Xin Huang et al. apresentam um protótipo de segurança para ambientes IdC baseando-se em cenários onde a IdC está presente regularmente (Empresas, Casas e Hotéis). Este protótipo, nomeado de *SecIoT*, aplica os mecanismos de segurança apresentados como indispensáveis na proteção das aplicações IdC e na da privacidade do utilizador, faz com que sejam entendidos os vários problemas de segurança em ambientes IdC, assim como fornecer algumas soluções eficazes. Portanto, pensando nas experiência de privacidade do utilizador, os autores aprofundam os seus estudos e procuram de forma estratégica saber das necessidades dos próprios utilizadores através de inquéritos nos cenários acima citados, quais os requisitos e mecanismos de segurança essenciais e indispensáveis para garantir tanto a privacidade do próprio utilizador e manter seguro os dispositivos inteligentes ligados à IdC.

Os autores E. Vasilomanolakis, J. Daubert et al. [VDL+16] abordam de forma abrangente as propriedades de segurança no que diz respeito a arquiteturas exclusivas para ambientes IdC, e através de um estudo aprofundado aos desafios que este ecossistema apresenta relativamente à segurança e à privacidade, são também analisados os requisitos de segurança mais dominantes na IdC. Requisitos esses que devem ser tidos em conta para garantir maior consistência defensiva a ataques, e melhor segurança dos dados partilhados pela rede. Para isso, os autores comparam as arquiteturas existentes e analisam as brechas de segurança em relação às propriedades e, através de um mapeamento com os requisitos de segurança, é avaliado nível de segurança que essas arquiteturas possuem de acordo aos cenários em que são utilizados. Todavia, a ineficiência no emprego dos mecanismos seguros para ambientes IdC acarretam vários problemas a este ecossistema.

Em [PHV17] os autores S. Pal et al. propõem vários requisitos de segurança específicos para a IdC e através dessa proposta apresentam um mecanismo de controlo de acesso onde são usados o *Attribute-Based Access Control* (ABAC), responsável pela definição dos direitos de acesso aos utilizadores através de políticas que combinam vários atributos e o *Role-Based Access Control* (RBAC), responsável por empregar funções predefinidas que carregam um conjunto específico de privilégios associados aos utilizadores, além de abordarem os problemas de segurança e ataques direcionados a esse sistema. No entanto, como um mecanismo segurança leve, este mecanismo não resolve todos os problemas de segurança inerentes à IdC. Contudo, os autores garantem que, ao adotar os requisitos abordados e o mecanismo de segurança proposto, seguindo a abordagem apresentada pelo artigo, é possível conseguir maior segurança, impedindo de forma eficaz as intenções maliciosas dos atacantes.

N. Kaliya e M. Hussain propõem, em [KH17], um gestor de segurança que assegura a privacidade dos dados do utilizador através de um mecanismo de Controlo de acesso que traz consigo o conceito de *Access Control List* (ACL) para ambientes IdC. Em paralelo a este gestor de segurança, é proposto ainda um outro mecanismo de controlo de acesso que incorpora um mecanismo de autenticação com certificados digitais e um mecanismo de criptografia de chave secreta. Portanto, esta proposta destaca os principais desafios relacionados com a privacidade do utilizador nos ambientes da IdC e acrescenta ainda a necessidade de haverem mais projetos que exploram a fraca segurança nestes ambientes e consequentemente na criação de mecanismos de segurança que garantam maior tranquilidade aos utilizadores.

Por outro lado, existem também estudos e trabalhos relacionados aos testes de segurança e de funcionamento realizados em vários dispositivos com capacidades computacionais reduzidas e limitadas usados em ambientes IdC, com a finalidade de perceber até que ponto esses dispositivos são eficientes e eficazes no que concerne à segurança e à privacidade dos dados. Assim, esses trabalhos foram analisados e referenciados nesta dissertação (de seguida) para servirem de ajuda para a realização de um dos objetivos propostos, de acordo o tema em desenvolvimento.

Em [MRON17], os autores apresentam um estudo sobre segurança em dispositivos da IdC, tendo como objetivo melhorar a segurança dos dispositivos e garantir que as informações confidenciais do utilizador são protegidas através de mecanismos de segurança. Foram propostos três algoritmos de criptografia para o efeito, nomeadamente o AES, RSA e o *Triple Data Encryption Algorithm* (TDES) mas, na prática apenas o AES e o RSA foram usados. Os autores estabelecem uma comparação entre os mesmos algoritmos através de testes experimentais de funcionalidade, tempo de execução (velocidade), e o uso da *Central Processing Unit* (CPU) respetivamente. Os testes foram realizados num dispositivo como Raspberry Pi. Os resultados mostraram, segundo os autores, que nos testes de tempo de execução (velocidade), o AES teve melhor desempenho, garantindo melhor resposta as aplicações em IdC no qual são utilizadas; no teste do uso da CPU, e sabendo que o uso dos algoritmos consomem muito processamento da CPU, mostraram que o algoritmo AES consome menos CPU em relação ao RSA, sendo uma boa escolha para proteger aplicações em IdC. Os autores concluem que, em função dos testes realizados, o algoritmo de criptografia AES é o que melhor se adapta aos objetivos do estudo realizado e chega a ser a melhor opção para a proteção dos dispositivos em IdC.

Lukas Malina et al. [MHFH16] apresentam os resultados obtidos a partir de testes de desempenho, realizados em plataformas para IdC através de algoritmos criptográficos, nomeadamente o AES, o SHA, o RSA, o *Elliptic Curve Diffie–Hellman* (ECDH), o *Elliptic Curve Digital Signature Algorithm* (ECDSA) e ainda funções que geram números aleatórios. Assim como os autores Gift Matsemela et al., em [MRON17], os resultados mostraram que o uso do algoritmo criptográfico AES e o uso de funções geradoras de números aleatórios em micro-controladores são bastantes eficientes, sobretudo quando comparadas com os outros algoritmos criptográficos mencionados no artigo, e podem ser implementadas em aplicações da IdC e executadas em ambientes ou plataformas com restrição de recursos.

Dimas Dwiki Ismoyo e Rini Wisnu Wardhan em [IW17] realizaram um estudo de pesquisa comparativa em redes *Virtual Private Network* (VPN) entre diferentes algoritmos de cifra de chave contínua e algoritmos de cifra por bloco, nomeadamente o ATHS3 e o *Vast Encryption Algorithm* (VEA) respetivamente, usando um SBC Raspberry Pi Model B+ como o *gateway* da VPN.

Além do uso dos algoritmos de cifra, um outro objetivo apresentado pelos autores é a escolha do algoritmo mais eficiente para a proteção dos dados que são enviados pela VPN, através da análise do desempenho da taxa de transferência de dados e o uso da memória de cada algoritmo no *gateway*. Assim sendo, os resultados obtidos dos testes realizados no SBC Raspberry Pi Model B+, a partir de ficheiros com capacidades de 1 Mb a 1 Gb mostraram que o algoritmo de cifra de chave simétrica contínua ATHS3 preenche os objetivos do estudo por possuir um melhor desempenho em termos de transferência de dados e ser mais eficiente, com um valor próximo a 1,07 vezes e uso menor da memória de 3.816 KB, em comparação com os 3.888 KB do VEA no *gateway* VPN.

O trabalho apresentado por Stig Tore Johannese em [Joh14] analisa o desempenho de um dispositivo Arduino due a executar o algoritmo criptográfico AES com uma chave de 128 bits e estabelece uma comparação de velocidades entre os modos de cifra ECB e CTR. Por causa das limitações de hardware do Arduino, foram usados ficheiros com pequenas quantidades de dados. As especificações de hardware do Arduino são um chip baseado em ARM Cortex-M3 de 32 bits, de 84MHx e 512 Kb de memória. Para o servidor foi usado um computador Lenovo T61. Foram gerados e usados 10 ficheiros aleatórios de 10 KB, sendo executados num total de 1023 experiências para cifragens no modo ECB, e num total de 1014 vezes para cifragens no modo CTR. O trabalho mostra que o desempenho do AES no modo ECB com uma taxa de transferência de 28.719 kiB/s, não foi tão rápido em relação ao modo CTR com a taxa de transferência de 241.493 kiB/s.

2.3.2 Comparação dos Trabalhos Relacionados

Ao serem analisados de forma exaustiva os artigos e trabalhos relacionados percebe-se que poucos são os artigos relacionados ao tema em desenvolvimento, muito embora estejam todos relacionados com a segurança quer dos dispositivos, dos dados, assim como do próprio ambiente em si. Nesta secção apresenta-se uma tabela (Tabela 2.2) comparativa destes artigos baseando-se em algumas questões inerentes à segurança e à privacidade da IdC. Esta comparação é baseada em questões como: definição e desafios da IdC de modo abrangente, segurança dos dispositivos (e.g., vulnerabilidades, riscos e ameaças), requisitos e mecanismos de segurança, e por último o mapeamento destes requisitos. Foram usados três(3) símbolos para se ilustrar se determinada questão é abordada no trabalho em análise ou não. O símbolo ✓ significa que o artigo apresenta detalhes de forma explicita sobre o assunto, o símbolo ✗ significa que o artigo não aborda a secção especificada na tabela, e o símbolo ✧ significa que o assunto é mencionado no artigo, mas sem muitos detalhes.

Tabela 2.2: Comparação dos trabalhos relacionados e abordados nesta dissertação.

Autores e Artigos	Objetivos	Ano	Definição e desafios	Segurança dos dispositivos	Testes criptográficos	Requisitos de Segurança	Mecanismos de segurança	Mapeamento Requisitos de segurança
Huang et al. [HCLY16]	Framework para segurança em IdC.	2016	✓	✗	✗	✓	✧	✗
Akinbi et al. [AP15]	Segurança em modelos de nuvem PaaS.	2015	✧	✓	✗	✓	✗	✗
Tae et al. [ATANM+16]	Requisitos de segurança nos sistemas moveis de saúde.	2016	✧	✓	✗	✓	✓	✓
Lin et al. [LYZ+17]	Segurança na IdC.	2017	✓	✓	✗	✓	✓	✗
Jing et al. [JVW+14]	Segurança em ambientes inteligentes.	2014	✓	✓	✗	✓	✧	✗
Ye et al. [YQ17]	Arquitetura de segurança para dispositivos na IdC.	2017	✓	✓	✗	✓	✓	✗
Kim et al. [OK17]	Requisitos de segurança para IdC.	2017	✓	✓	✗	✓	✓	✗
Kim. [Kim15]	Requisitos de segurança para sistemas em IdC.	2015	✓	✓	✗	✓	✧	✗
Hossain et al. [HFH15]	Desafios de segurança para IdC.	2015	✓	✓	✗	✓	✓	✧
Mendez et al. [MPY17]	Segurança e privacidade na IdC.	2017	✓	✓	✗	✓	✓	✧
Daubert et al. [VDL+16]	Mecanismos de segurança baseado em framework para IdC.	2017	✓	✓	✗	✓	✓	✓
Kaliya et al. [KH17]	Mecanismos de segurança para IdC.	2017	✓	✓	✓	✓	✓	✗
Gift et al. [MRON17]	Integridade da IdC.	2017	✧	✓	✓	✓	✓	✗
Lucas et al. [MHFH16]	Soluções para a segurança e privacidade na IdC.	2017	✓	✓	✓	✧	✧	✗
Ismoyo et al. [IW17]	Desempenho de algoritmos criptográficos.	2017	✧	✓	✓	✧	✧	✗
Stig. [Joh14]	testes de segurança.	2014	✧	✓	✓	✧	✧	✗
Dissertação	Mapeamento dos Requisitos de segurança em IdC	2018	✓	✓	✓	✓	✓	✓

Como é apresentado na tabela 2.2, é visível que o leque de trabalhos abordados estão num intervalo de três anos (compreendidos entre o ano de 2014 a 2017). Entretanto, como foi mencionado acima, esta é uma área não tão desenvolvida, e que ao poucos vai ganhando interesse da parte dos investigadores desta área e, estes vão contribuindo com estudos aprofundados e com a criação de mecanismos eficazes e eficientes para garantir que a IdC seja formada por coisas seguras, tendo em conta as propriedades de segurança e as regras de comunicação entre os dispositivos M2M. Um outro aspeto importante e que não pode ser ignorado, é que mesmo com os desafios apresentados na subsecção 2.2.2, a IdC tem ocupado o seu espaço e tem demonstrado que é uma tecnologia que veio para não só melhorar as atividades pessoais, empresariais e industrias, como para ser a próxima revolução da Internet.

2.4 Principais Problemas de Segurança na Internet das Coisas

Os dispositivos da IdC apresentam uma debilidade de segurança e isso representa riscos para os utilizadores e para o bom funcionamento da Internet. Estes riscos continuam a aumentar com base no uso da mesma quer da parte dos utilizadores individuais, como da parte de empresas e indústrias.

Relativamente aos serviços prestados na IdC por empresas focadas no desenvolvimento assim como na expansão desta tecnologia, a segurança tem-se tornado um ponto sensível nos seus provedores de serviços na Internet, pois quanto maior for o crescimento da IdC, maior será também a preocupação em relação à segurança dos serviços prestados. Neste sentido, a IdC tem incentivado a criação de medidas de segurança, pese embora que não tem sido suficiente para colmatar todos os problemas de segurança que têm vindo a surgir neste ecossistema. Entretanto, devido às vulnerabilidades, a tendência é que as ameaças à segurança e a privacidade cresçam, assim como as formas de ataques voltadas para os dispositivos da IdC. Qualquer aparelho inteligente pode ser ligado a Internet e ser controlado através de um computador ou *smartphone*. Embora os dados da IdC advindos dos sensores e objetos inteligentes possam ser coletados e processados com a intenção de melhorar o nosso dia a dia, as comunicações entre dispositivos inteligentes também podem revelar informações particulares sobre cada utilizador.

2.4.1 Riscos para os Utilizadores

No contexto das *residências inteligentes*, por exemplo, dificilmente são encontrados mecanismos fortes de segurança. Um utilizador malicioso pode aproveitar estas vulnerabilidades e apoderar-se do sistema presente na residência. No caso da IdC todos os dispositivos inteligentes de uma casa (e.g., TV, Rádio, Micro-ondas, frigorífico, portas, janelas, lâmpadas etc.) ligados ao sistema residencial representam riscos e ameaças tanto para os demais dispositivos na rede, como para os utilizadores do sistema. Dois dos maiores problemas inerentes a estes riscos são a falta de capacidade de atualização do software e o fraco suporte técnico.

No setor da *saúde*, os riscos e ameaças provenientes da fraca segurança encontrada nos dispositivos da IdC voltados especificamente para este sector representam, em geral, maior preocupação em comparação com os demais cenários, tudo porque estes dispositivos estão diretamente associados a um aspeto crítico da vida das pessoas, podendo em alguns casos ser uma questão de vida ou morte. Por exemplo, investigadores descobriram que é possível invadir dispositivos cardíacos implantáveis, podendo esgotar a bateria ou administrar ritmos ou choques incorretos, causando a morte do paciente [Lar17]. De igual modo, foi descoberta uma vulnerabilidade de segurança numa bomba de insulina através da qual um atacante poderia fazer com que a mesma administrasse aquela hormona numa dose acima do normal [REU16]. Portanto, é indispensável a implementação de mecanismos de segurança em dispositivos médicos da IdC, pois, do mesmo modo que estes dispositivos são usados para salvar vidas, podem ser usados para tirar.

No sector dos *transportes*, os benefícios associados ao novo paradigma têm também sido notados, por exemplo porque a quantidade de dispositivos interligados fornecem enormes volumes de dados referentes ao fluxo de tráfego, reduzindo sistematicamente o congestionamento de trânsito. É claro que, por esses motivos, as infraestruturas de transportes inteligentes necessitam de soluções abrangentes de segurança. Por outro lado, atualmente, já se vêm empresas a investirem em veículos suficientemente inteligentes e equipados com dispositivos da IdC sem

os mecanismos apropriados de segurança. Em alguns casos foi demonstrado que estes veículos poderiam ser controlados remotamente, mantidos como reféns por invasores ou mesmo usados em ataques de *Distributed Denial of Service* (DDoS). Tecnologias avançadas como *Secure Booting* tomam especial relevo na IdC e neste setor, e garantem que a integridade dos veículos não é violada tão facilmente.

2.4.2 Riscos para a Internet

A Internet é hoje utilizada por milhões de utilizadores, das mais variadas idades e culturas, com potencial de vir a ser considerada como um bem essencial no futuro. Qualquer possibilidade de disrupção ao seu funcionamento, *per se*, tem o potencial de destruir a confiança pública e o papel crescente da rede das redes na economia e na sociedade em geral. Essa dinâmica põe em risco os benefícios de uma sociedade ligada a ambientes inteligentes, o comprometimento cívico, o comércio digital e a produtividade. A segurança é por isso fundamental, e deve ser uma responsabilidade compartilhada para garantir o crescimento contínuo dos benefícios de longo alcance da Internet.

2.4.3 Ameaças e Ataques Emergentes na Internet das Coisas

Os utilizadores de dispositivos da IdC podem não notar ou sequer se importar se os seus equipamentos estão a ser usados para fins maliciosos. Contudo, uma vez que um atacante tenha domínio sobre um dispositivo num ambiente da IdC, ele pode comprometer todos os outros equipamentos que estão conectados ao mesmo ambiente. A seguir são apresentados alguns dos ataques muito populares recentemente dirigidos a cenários inteligentes voltados para a IdC.

Denial of Services *(DoS) e DDoS:* os ataques de negação de serviço, embora muito frequentes em dispositivos da IdC, existem há muitos anos no ecossistema da Internet. Atualmente, estes ataques são normalmente levados a cabo através de uma rede de dispositivos comprometidos (e.g., computadores, *smartphones*) conhecida como *Botnet*, cujo papel é de executar e ampliar um ataque dirigido a um servidor ou serviço, com a finalidade de o deixar indisponível para os demais utilizadores. As consequências de um ataque DDoS bem-sucedido podem ser catastróficas, em particular para as empresas que realizam as suas atividades a partir da Internet, com impacto negativo na imagem e economia da mesma (e.g., prejuízos financeiros, aborrecimentos por parte dos consumidores e perda de produtividade, reputação prejudicada, e perda no controlo dos dados). Proteger um ambiente inteligente de um ataque DDoS é algo difícil para a maioria das situações; isto porque é preciso ter recursos necessários para mitigar o possível ataque ou porque, em alguns casos, não é possível mitigá-los.

Man-in-the-Middle: este modelo de ataque comporta frequentemente todas as ações maliciosas relacionadas com escuta, interrupção ou violação das comunicações entre dois dispositivos separados. Ou seja, é um ataque onde o invasor secretamente interceta ou transmite mensagens entre duas partes quando estas estão convictas de que estão comunicando diretamente umas com as outras. No entanto, como o atacante tem a comunicação original, pode levar os destinatários a pensar que ainda estão a receber uma mensagem legítima. Estes ataques podem ser extremamente perigosos no mundo da IdC, devido à potencial e enorme quantidade de informações que a rede pode vir a partilhar.

Ransomware: é um software malicioso que, após a infeção, mantém cativos os arquivos no sistema ou a própria máquina da vítima até que um pagamento seja efetuado. Os ataques de

Ransomware estão atualmente em constante melhoria, com os programadores a tentar dificultar cada vez mais o desenvolvimento e o planeamento de métodos eficazes de prevenção. Basicamente existem dois tipos de *Ransomwares:* o *Crypto Ransomware e o Locker Ransomware*. O primeiro é responsável apenas por cifrar os arquivos da vítima (e.g., documentos, textos, fotos, vídeos etc.) e exigir um resgate em dinheiro. O segundo é responsável por bloquear apenas o sistema da vítima (impossibilitando a mesma de lhe aceder), deixando os arquivos intactos, mas também exigindo um resgate em troca. Alguns exemplos comuns destes ataques são os *Ransomwares NotPetya, WannaCry, e Locky*. Os ataques recorrendo a/culminado na instalação de *ransomware* têm-se espalhado rapidamente e não apenas em utilizadores comuns da Internet, mas também em ambientes sofisticados de rede convencionais, assim como em ambientes da IdC.

Malwares: tal como em redes tradicionais, tem-se notado um crescimento notável de *malware* em dispositivos da IdC. O motivo para o crescimento destas ameaças é que a IdC é frágil e está exposta aos atacantes dado que em muitos casos os *firmwares* dos equipamentos não são atualizados constantemente e não têm qualquer proteção de segurança. Este tipo de ameaça pode até ser menos danoso em comparação com *ransomware* mas, ainda assim, é de extrema importância a implementação de mecanismos eficazes de segurança nestes dispositivos, pois diariamente lidam com informações sensíveis dos utilizadores. Muitos dos *Malwares* existentes na IdC estão voltados para ataques de DDoS, como é o caso do *Malware Mirai*, que efetua uma busca exaustiva por equipamentos na IdC que estejam vulneráveis para controlá-los.

2.5 Conclusão

Um dos principais objetivos da IdC, é permitir a interligação de objetos inteligentes do mundo real com o mundo virtual por meio de sensores, e com isso, agilizar a partilha de informações de forma ubíqua, mais transparente e útil para a vida humana em geral. Percebe-se ainda que a IdC tem vindo a crescer dia após dia, tornando-se cada vez mais sólida em termos de estrutura e em termos de alcance. Por outro lado, tem sido aplicada em vários setores chave da sociedade (e.g., saúde, educação, desporto, transporte), e várias empresas têm apresentado um crescimento absurdo relativamente aos lucros explorando este paradigma de alguma forma.

Neste capítulo, foram abordados alguns conceito intrínsecos da IdC, desde as suas principais características até aos seus principais desafios, sendo que esses desafios representam grandes riscos para os utilizadores, empresas e a sociedade em geral. Foram analisados ainda, de forma abrangente, alguns dos trabalhos com soluções específicas para os problemas de segurança e privacidade que este ecossistema apresenta, assumindo que estes tiveram uma parte bastante importante na elaboração desta dissertação. Na parte final do presente capítulo é apresentada a explicação dos meios de comunicação existentes entre os dispositivos da IdC, assim como a sua arquitetura em camadas, com o objetivo de dar a entender a forma como estes objetos comunicam entre si. É de salientar que a exposição feita neste capítulo é indispensável para se compreender, no próximo capítulo, os mecanismos usados para combater a fraca segurança e as ameaças enfrentadas pelos ambientes inteligentes, e mapear estes mecanismos aos requisitos de segurança aplicáveis na IdC.

Capítulo 3

Requisitos e Mecanismos de Segurança em Internet das Coisas

3.1 Introdução

Á segurança na IdC tem sido motivo de preocupação para os utilizadores principalmente por estar ainda na sua infância, e por não existirem mecanismos fortes e adaptados a partes da arquitetura que protejam na íntegra tanto o ambiente como os utilizadores. No entanto, através dos estudos e pesquisas realizadas nesta área, vários mecanismos estão a ser desenvolvidos para solucionarem estes e outros problemas de segurança. Com este objetivo em mente e neste capítulo, os requisitos de segurança importantes para a IdC são apresentados de forma sucinta, também com o intuito de fornecer um maior vislumbre da necessidade de segurança que este ecossistema carece. Este capítulo descreve os requisitos e mecanismos de segurança importantes para IdC; um dos aspetos importantes desta dissertação é o mapeamento destes requisitos de segurança para ambientes da IdC. A secção 3.2 lista e aborda os requisitos de segurança e a secção 3.3 lista e aborda de forma sucinta os mecanismos de segurança apropriados para IdC.

3.2 Requisitos de Segurança

Esta secção analisa os requisitos de segurança normalmente descritos para a segurança da informação. Esses requisitos também são tipicamente aplicáveis na segurança da IdC. Os requisitos de segurança são conjuntos de regras ou propriedades que orientam o funcionamento seguro das operações em vários setores como transporte, saúde, construção, etc. Na IdC, esses requisitos são utilizados no desenho de mecanismos seguros voltados para este ecossistema. Assim sendo, os princípios básicos da segurança da informação são representados pela tríade conhecida por Confidencialidade, Integridade e Disponibilidade [SKB+08], que atualmente regem os objetivos, o planeamento e a implementação da segurança em sistemas ou informações sigilosas, com o intuito de serem protegidos. No entanto, as propriedades de segurança acima mencionadas foram expandidas e especializadas ao longo do tempo para incluir outras propriedades de segurança, como se descreve a seguir. Portanto, é importante compreender os requisitos de segurança e aplicá-los no sentido de se perceber quais os mecanismos de segurança que satisfazem esses requisitos.

3.2.1 Confidencialidade

A confidencialidade garante que os dados estejam apenas disponíveis para utilizadores apropriados, ou seja, apenas para os utilizadores autorizados. Assim, em IdC, é importante garantir que os dados de carácter sigiloso, recolhidos por dispositivos em ambientes inteligentes, não caiam em mãos oportunistas e sejam usados de forma errónea [LYZ+17]. Este tema é em muitos casos analisado de forma superficial. Com a quantidade de dados que circulam na Internet, hoje em

dia, há cada vez mais pessoas que investem nas formas de obter estes dados a qualquer custo.

A confidencialidade representa uma questão fundamental em cenários da IdC, indicando a garantia de que somente entidades autorizadas podem aceder e modificar dados. Isto é particularmente relevante no contexto de negócios, em que os dados representam um ativo a ser protegido para salvaguardar a competitividade, valores de mercado, e a reputação da empresa. No contexto da IdC, tanto os utilizadores como os objetos autorizados, podem estar capacitados para aceder a dados confidenciais, mesmo os separados por níveis de acesso. No entanto, é importante abordar dois aspetos: primeiro a definição de um mecanismo de segurança voltado para a proteção dos dados (e.g., controlo de acesso, criptografia) e, segundo, a definição de um processo de autenticação (com um sistema de gestão de identidades).

Para se entender a importância da confidencialidade nos sistemas da IdC, considere-se o seguinte exemplo: um sistema inteligente de monitorização do clima ambiental, em que os dados são usados para fornecer um conjunto de alertas antecipado contra possíveis terramotos. Neste cenário, os dados devem apenas ser acessíveis pela autoridade específica, que é a responsável por implementar regras e estratégias de gestão de riscos. No entanto, o vazamento de tais dados para a sociedade em geral causaria situações constrangedoras e de pânico, o que não é aconselhável quando existem eventos catastróficos do género.

3.2.2 Integridade

A integridade é essencial para os dados no contexto da IdC, pois significa que independentemente de como é a gestão do sistema do ambiente inteligente, estes dados devem permanecer inalteráveis quando em descanso ou transmissão. Significa que a alteração dos dados deve ser apenas realizada por utilizadores autorizados e devidamente instruídos. No entanto, em muitos casos, devido ao fraco foco em aspetos de segurança, pessoas não autorizadas podem com pouco esforço conseguir obter estes dados (e.g., dados pessoais, dados bancários, dados organizacionais, governamentais), roubá-los ou até mesmo alterá-los de acordo as suas necessidades.

Estes dados, que são de carácter importante para os utilizadores e instituições, podem estar expostos facilmente sem uma medida de segurança que vai contra as várias ameaças do ponto de vista do valor que estes dados representam para os utilizadores. Mecanismos que verifiquem se os dados sofreram alguma alteração são necessários neste sentido.

A verificação da operação correta e segura de sistemas da IdC requer também um mecanismo eficiente para verificar coletivamente a integridade não só dos dados mas também a do software e de todos os dispositivos ligados ao ambiente IdC, a fim de detetar alterações de software mal-intencionadas e não intencionais. A integridade pode ser conseguida usando mecanismos (ou combinações de) como: criptografia, controlo de acesso, assinatura digital, funções de *hash*, Código de Autenticação de Mensagem (MAC) e atestação.

3.2.3 Disponibilidade e Conformidade

A disponibilidade refere-se à acessibilidade que os utilizadores têm em relação aos dados e em sistemas da IdC. A necessidade de se ter informações fidedignas conduz-nos ao uso constante de meios tecnológicos que permitem a busca de dados através da Internet. Os dispositivos da IdC partilham diariamente dados em quantidades potencialmente elevadas. No entanto, decidir quem deve ter acesso as estes dados torna-se bastante importante, uma vez que os

dados são partilhados numa rede mundial onde qualquer pessoa com conhecimento lhes pode aceder [ADAS17].

A disponibilidade é o requisito que determina que o acesso a dados é possível a utilizadores com permissões de acesso específicos. Em empresas que fazem o uso da IdC, este requisito muitas vezes é dividido em níveis e sub-níveis no intuito de permitir que os dados não estejam disponíveis para todos os funcionários, mas que cada funcionário tenha apenas informações segundo a área em que está inserido. A disponibilidade pode ser comprometida por falta de um plano de recuperação de dados, ou seja, os dados podem não estar disponíveis devido a problemas como desastres naturais (e.g., enchentes, terramotos, incêndios etc.) e problemas técnicos. Esta propriedade pode ser garantida por mecanismos de segurança controlo de acesso, cópias de segurança e autenticação, e também por mecanismos de redundância de dados.

3.2.4 Autenticidade e Confiança

Quando lidamos com pessoas, uma propriedade que se torna indispensável é a confiança. Transmitir confiança é algo importante para as relações humanas para que estas sejam desenvolvidas sem sobressaltos. As empresas por exemplo, devem conquistar e ganhar a confiança dos seus clientes para que haja da parte dos clientes maior interesse nos produtos ou serviços prestados pelas empresas. Este mesmo conceito é utilizado quando manuseamos dispositivos conectados à Internet. Os dispositivos em IdC podem ser inteligentes mas precisam sempre de uma mão humana para o correto funcionamento. No entanto, é um desafio interessante pensar como garantir que os dados e os dispositivos são confiáveis.

Entretanto, um dos principais objetivos de uma infraestrutura de rede como a IdC é assegurar que a comunicação realizada entre os dispositivos acontece de forma segura e confiável utilizando diferentes protocolos e parâmetros de comunicação existentes. A confiança nos sistemas IdC poderia ser alcançada com a implementação de mecanismos de segurança confiáveis em todas as camadas da sua arquitetura, com maior ênfase nos protocolos de comunicação da camada de aplicação, garantindo maior confiança na partilha de dados pela rede, e garantindo também maior nível de fiabilidade nas aplicações IdC através dos mesmos protocolos.

Por outro lado, no contexto das comunicações M2M, a autenticidade garante que a fonte da transmissão dos dados através dos dispositivo IdC seja verdadeira. De outro modo, que a fonte de transmissão seja na realidade aquela que afirma ser, e que os dados durante a transmissão não sofram alterações indevidas. Assim sendo, a autenticidade estabelece uma ponte de confiança entre o remetente e o destinatário, neste caso, e visto que os dados na IdC são usados para diversos fins, em particular em processos de tomada de decisão, torna-se importante a implementação de mecanismos que suportem essa propriedade de segurança. A autenticidade pode ser garantida por primitivas de criptografia ou mecanismos de controlo de acesso.

3.2.5 Privacidade

A segurança e a privacidade são aspetos primordiais quando nos conectamos com o mundo através da Internet, quando se partilham informações a nosso respeito. Estar atento e precavido é um ato de inteligência hoje em dia, num mundo em que é frequentemente concluído que não existe sistema informático que garanta segurança a 100%. Esta conclusão tem sido corroborada pelo que os jornais têm transmitido sobre as fugas de informações confidenciais e ataques *hackers* qualificados a instituições de renome. Em [SRGCP15], são apresentados alguns projetos

referentes à privacidade em IdC que estão em curso e que fornecem métodos para fortalecer os mecanismos de segurança já existentes e muito conhecidos, mas pouco usados em dispositivos da IdC por parte dos fabricantes.

É importante que haja consentimento entre os provedores de serviços e utilizadores (consumidores) pela colheita e partilha de dados sensíveis. Este consentimento deve basear-se na confiança mútua entre as partes envolventes, devendo os provedores explicar com clareza aos utilizadores quais os dados recolhidos e para que objetivos eles vão ser usados. Assim, o princípio da privacidade exige que os utilizadores tenham a plena noção dos dados que estão cedendo aos serviços, e estabelece que os utilizadores mantenham o controlo total sobre os seus dados. Entretanto, não há dúvidas que para a IdC, não obstante todos os seus aspetos positivos, a implementação de tais tecnologias não pode ser liderada por grupos de pessoas que são motivados apenas pela comercialização da mesma, pois como tem sido notado, existem mais políticas de comércio a rodear a tecnologia e o paradigma do que propriamente resoluções de problemas a si inerentes.

Sendo a privacidade um direito humano, é interessante ressalvar que já existem políticas que regularizam a recolha de dados dos utilizadores por parte de empresas ou serviços, nomeadamente na União Europeia. Embora não seja global, o regulamento começa aos poucos a ser adotada por empresas ao redor do mundo, servido esta medida, por si, como um mecanismo de segurança para o próprio utilizador. Os utilizadores estarão cada vez mais conscientes das implicações da privacidade deste nível de conectividade através da interação com a IdC. Contudo, para assegurar a privacidade, deve-se aumentar a transparência em relação ao que é feito aos dados, fortalecer a sua supervisão, e empregar a privacidade através de mecanismos de segurança ideais para ambientes inteligentes.

3.2.6 Autorização e Autenticação

A Autorização e Autenticação, conhecidas como parte do controlo de acesso [LXC12], são responsáveis por conceder permissões aos utilizadores dos dados ou acesso a informações confidenciais. Na IdC, é necessário que estas propriedades estejam habilitadas para que os serviços estejam simplesmente disponíveis para os utilizadores registados. No entanto, a confiança existente entre os dispositivos é de extrema relevância. A autenticação garante que a comunicação seja feita de forma segura e confiável. Mecanismos de controlo de acesso são indispensáveis na IdC pois garantem o devido acesso aos serviços e recursos disponíveis na rede. O uso correto destas propriedades de segurança na IdC é essencial para minimizar o acesso não permitido a informações que se deseja proteger.

3.2.7 Não-Repúdio

O Não-Repúdio é a propriedade de segurança que garante que nem o remetente nem o destinatário possam negar a participação numa determinada ação. Com a evolução e o crescimento da IdC, a quantidade de empresas que prestam serviços e disponibilizam os seus recursos na nuvem para IdC procuram estabelecer mecanismos necessários para uma maior adoção dos serviços e atender de forma eficaz e eficiente as necessidades dos utilizadores. No entanto, o uso da propriedade do não-repudio é crucial para se saber na realidade quem é o responsável por atividades que comprometam ou não o bom funcionamento dos dispositivos assim como do acesso a informações importantes e confidenciais, eventualmente em processos posteriores de auditoria e apuramento de responsabilidades, se necessário.

3.3 Mecanismos de Segurança

Como analisado na secção 3.2, existe uma forte necessidade em garantir que os dispositivos inteligentes com capacidades de processamento reduzidas estejam seguros quando conectados à Internet. Vários estudos têm sido realizados em torno da segurança IdC como apresentado em [KMAM16, AOHA17]. Contudo, as implementações efetivas e eficientes têm na realidade sido poucas. Infelizmente, muitos métodos de segurança tradicionais não podem ser imediatamente transportados para sistemas IdC, o que provoca um vazio e abre a superfície de ataque tanto para ambientes da IdC, como para a Internet de uma forma geral. Para a proteção contra esses ataques, é importante examinar os problemas de segurança de acordo com o fluxo de informação e potenciais pontos de ataques. A IdC estende a Internet ao mundo físico e traz com isso novas variáveis ao problema.

Dada a complexidade que a IdC apresenta e a necessidade de implementar funcionalidades de segurança em diferentes lugares e de diferentes formas, um único mecanismo não pode fornecer uma solução de segurança completa nem abordar todas as ameaças e ataques. Como exemplo, a criptografia será necessária tanto para dispositivos individuais quanto para aplicações. Dentro das limitações de segurança da IdC, ainda é possível projetar mecanismos de acordo com os requisitos que identificamos e ainda, monitorizar, analisar e melhorar, onde necessário, para garantir que os objetivos da IdC sejam alcançados sem trazer aos utilizadores tantas preocupações como existem atualmente. Portanto, para se dar continuidade aos objetivos deste projeto de dissertação, esta secção aborda de forma sucinta alguns dos mecanismos de segurança utilizados na proteção de dispositivos e informação, e estabelece uma ligação com os requisitos de segurança abordados na secção 3.2.

3.3.1 Atualizações Autenticadas

No mundo em que o esforço humano é reduzido pelo uso de tecnologias e ferramentas inteligentes (dispositivos, objetos), na sua maioria conectadas à Internet, manter tais ferramentas atualizadas é crítico, pois as atualizações trazem normalmente melhorias de *firmware* ou *software*, normalmente em termos de eficiência da produtividade do mesmo, mas também em termos de segurança. Diariamente, são desenvolvidos códigos maliciosos com a finalidade de explorar as vulnerabilidades em dispositivos da IdC. Assim, a disponibilidade e o uso de mecanismos seguros, como é o caso da utilização de *patches* de segurança por empresas que usam assinaturas digitais nos seus dispositivos da IdC, garantem que os dispositivos estão seguros contra ameaças conhecidas e garantem ainda maior confiança na execução de *software*, como abordado na subsecção 3.3.2. Especialistas em segurança e investigadores recomendam que todos os dispositivos da IdC estejam equipados com esse mecanismo. Embora existam muitos mecanismos proprietários de atualização de *firmware* em uso hoje, não existe uma abordagem moderna que permita atualizações seguras para *firmware* que possam ser usadas em dispositivos de diferentes fabricantes na IdC de uma forma generalizada. No entanto, manter os dispositivos sempre atualizados através de *patches* de segurança autenticada minimizam as vulnerabilidades existentes e dificultam a vida dos criminosos digitais.

3.3.2 Arranque Seguro (*Secure Booting*)

Os dispositivos da IdC, assim como os demais dispositivos ligados pela Internet, necessitam de uma abordagem de segurança que enfatiza a proteção tanto dos dados em si, a maneira que esses dados são acedidos, e como a proteção física da própria infraestrutura da IdC, pois, quanto

maior o risco apresentado por esses dispositivos, maior será a perda de dados. A segurança da IdC deve ser baseada numa estrutura de confiança, onde o sistema concorda que existe algum lugar seguro e confiável e que proporciona uma base segura para que o sistema possa ser carregado e inicializado com segurança, e para que um dispositivo em IdC seja executado de forma segura, o processo de arranque deve ser assegurado em primeira instância.

Assim sendo, *Secure Booting* é uma validação de segurança que é feita durante o processo de inicialização de um dispositivo [MAH16], que usa mecanismos de segurança como assinatura digital anexada à imagem do *software* para verificar a sua integridade e autenticidade e permitir que apenas o *software* autorizado, sem modificações, seja executado. *Secure Booting* é de extrema importância, pois previne e maximiza o grau de dificuldade para as situações em que alguém que pirateia um dispositivo, não possa encontrar uma maneira de substituir um ficheiro executável existente no dispositivo por um que contenha *malware*. Para os dispositivos sem *Secure Booting*, as funções perdem a confiança quando o sistema executa código malicioso, e na maioria das vezes isso acontece sem a perceção dos utilizadores. No entanto, com um processo de *Secure Booting*, as verificações no momento da inicialização do dispositivo devem identificar o arquivo inesperado e tomar as devidas ações corretivas.

3.3.3 *Firewall*

Uma *Firewall* é uma coleção integrada de medidas de segurança destinadas a prevenir o acesso não autorizado a um sistema informático em rede, e.g., dispositivos da IdC. A *firewall* monitoriza o tráfego de entradas e saídas de pacotes de dados de acordo com as políticas de segurança existente e concede permissão apenas aos pacotes considerados seguros. Uma *firewall* também evita que informações confidenciais sejam enviadas de um dispositivo sem a devida permissão, bem que como sejam emitidos comandos indevidos durante acessos remotos. Muitos dispositivos inteligentes necessitam de uma *firewall* ou de um sistema eficaz de inspeção de pacotes para controlar o tráfego. o Uso bem-sucedido de uma *firewall* dependerá da escolha do dispositivo apropriado, ou seja, o nível de segurança para os dispositivos inteligentes na IdC dependem das configurações do software de *firewall*, pois eles colocam uma barreira entre redes internas protegidas e redes externas confiáveis ou não, como a Internet [dKH17]. A figura 3.1 mostra uma ligação remota entre um utilizador e um dispositivo IdC através da Internet, com as *firewalls* ativas.

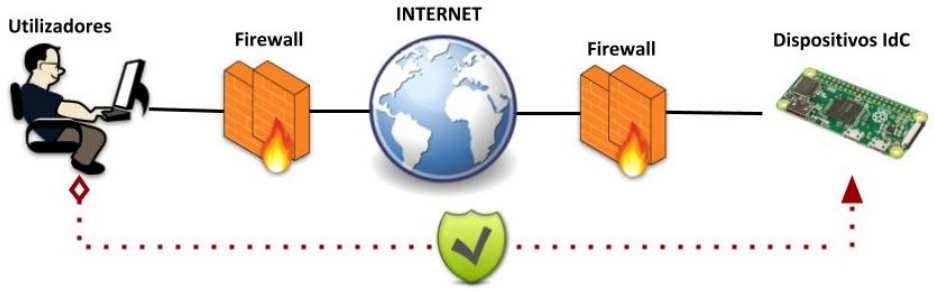

Figura 3.1: Ligação remota com dispositivos da IdC enfatizando possíveis posições de *firewalls* de rede ao longo do caminho.

Os dispositivos da IdC têm demonstrado possuir vários problemas de segurança desde o momento da criação até na comercialização e consequentemente a utilização. No entanto, até que as

questões de segurança sejam resolvidas, as empresas precisam ter maior cautela na criação de novos dispositivos inteligentes, e na disponibilização para os utilizadores. As *firewalls* têm sido bastante utilizadas e são vistas como um dos primeiros mecanismos de segurança que um dispositivo ligado à Internet deve possuir. Os utilizadores de dispositivos inteligentes comprometidos raramente percebem que os seus dispositivos são alvos fáceis e que podem ser usados para fins maliciosos. Contudo, a configuração adequada (e até seguindo uma política redutora de *bloqueia tudo exceto o explicitamente permitido* por defeito) de uma *firewall* pode evitar ou prevenir que os dados gerados por eles caiam em mãos erradas ou que lhes sejam feito ataques.

3.3.4 Criptografia, Certificação e Assinatura Digitais

A criptografia fornece hoje um conjunto de ferramentas cruciais para a segurança de sistemas informáticos, e as suas primitivas são usadas há décadas em comunicações bilaterais. Consequentemente, é indispensável para garantir também a segurança e a privacidade dos dispositivos e aplicações na IdC. Boa parte dos dispositivos IdC apresentam dificuldades na utilização de soluções criptográficas mais exigentes (e muitas das soluções existentes são exigentes computacionalmente). Deste modo, a escolha de algoritmos criptográficos seguros e *leves* torna-se o meio mais viável para garantir eficiência na segurança destes dispositivos. Os modelos criptográficos atuais e os esquemas de segurança são baseados em algoritmos de criptografia amplamente adotados em padrões de privacidade divididos em duas categorias: criptografia de chave simétrica, também conhecido como criptografia de chave secreta (a chave que é usada para cifrar, é a mesma usada para decifrar), da qual são exemplos os algoritmos DES, TDES, RC4 e o AES, e a criptografia assimétrica, também conhecida como criptografia de chave pública, da qual são exemplos os algoritmos RSA, *Digital Signature Algorithm* (DSA), e os algoritmos de cifra *Diffie-Hellman* (DH) e o *Elliptic Curve Cryptography* (ECC). Funções de *hash* seguras como a SHA são também primitivas importantes da criptografia moderna, tida em consideração nesta dissertação. Em geral, algoritmos que usam criptografia de chave secreta apresentam maior rapidez. No entanto, apresentam o problema da troca segura de chaves de cifra e não permitem assinaturas digitais sem nomeação de um agente de confiança. A figura 3.2 esquematiza de forma simplista o funcionamento da criptografia simétrica, assumindo que as chaves de cifra foram pré-estabelecidas.

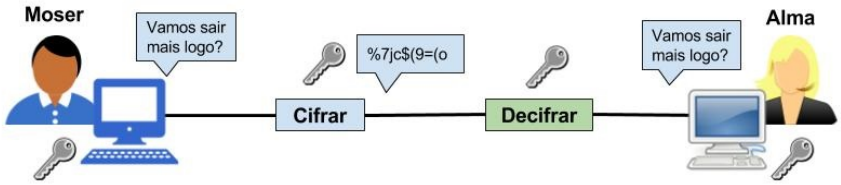

Figura 3.2: Esquema simplista de como funciona um algoritmo de cifra de chave simétrica.

A criptografia assimétrica usa duas chaves (privada e pública) para cifrar e decifrar, ou assinar e verificar a assinatura de uma mensagem. No entanto, a chave pública é do conhecimento de todos e é usada no ato de cifrar ou verificar uma assinatura digital, enquanto a chave privada é mais restrita e é usada para decifrar o criptograma ou fazer a assinatura digital, sendo que só o uso de uma chave maior (e.g., 2048 bits) no processo de criptografia torna o sistema seguro. No entanto, existem outros fatores que influenciam negativa ou positivamente a segurança e a privacidade dos dispositivos IdC, e.g., a forma como são guardadas as chaves privadas. A figura a seguir 3.3 apresenta, de forma simplista, o modo de funcionamento de uma cifra de

chave pública, denotando a chave privada com a cor preta e a chave pública com a cor cinza. O utilizador Moser usa a chave pública da Alma para cifrar uma mensagem que lhe é dirigida.

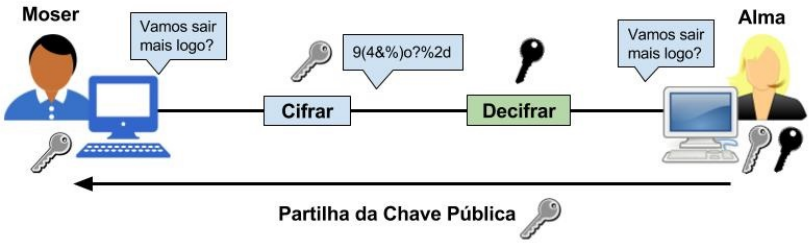

Figura 3.3: Esquema simplista de como funciona um algoritmo de cifra de chave pública.

Tanto no contexto da Internet ou sistemas legados, como no da IdC, é comum usar técnicas híbridas no contexto da criptografia. Estas técnicas usam mecanismos da criptografia de chave simétrica e da chave pública simultaneamente. E.g., um algoritmo de chave pública é usado para trocar chaves de cifra de sessão (simétricas) e os algoritmos de chave simétrica são usados para cifra de mensagens. A aplicação da criptografia na IdC está em fase de exploração no mundo da IdC.

Além daqueles mencionados antes, existem outros conceitos e mecanismos da criptografia moderna que podem ser associados à segurança nos dispositivos da IdC, nomeadamente certificados e assinaturas digitais, que desempenham um papel crucial no estabelecimento de propriedades e identidades, e na manutenção da integridade dos dados e dos próprios dispositivos. Os certificados também protegem os dados trocados entre dispositivos. Os certificados digitais são a base para estabelecer a confiança entre entidades, tanto na Internet como em redes privadas. São cada vez mais importantes para proteger dados e aplicações da IdC, estabelecendo formas de transmissão de confiança entre dispositivos e comunicações num ambiente inteligente. A manutenção da integridade dos dados e da privacidade nunca foi tão importante [SC15] como neste momento de expansão das redes de informação.

3.3.5 Controlo de Acesso e Gestão de Identidades

Muitos recursos importantes da IdC necessitam de proteção cuidadosa, incluindo a própria infraestrutura de rede, a parte física dos sistemas e a enorme quantidade de dados gerados por estes dispositivos em grande escala. O uso de mecanismos de controlo de acesso eleva os níveis de segurança, garantindo maior proteção dos dispositivos, pois são eles que medeiam o acesso a indivíduos, ou processos de acordo com políticas de segurança pré-estabelecidas. Assim sendo, a identificação, autenticação, autorização e não repúdio, abordadas na secção 3.2, são algumas das propriedades de segurança parcialmente preenchidas pelos mecanismos de controlo de acesso neste ecossistema. Quando bem definidos e implementados, protegem tanto a parte física da infraestrutura de rede, como a parte interna [OMEO17].

Um dos outros mecanismos que funciona em paralelo ao controlo de acesso é o da gestão de identidades. A garantia de que cada dispositivo e utilizador tem identidade que se pode distinguir e provar permite que as comunicações realizadas entre as entidades envolvidas, assim como a participação em vários serviços voltados para ambientes inteligentes, ocorra de maneira autenticada e autorizada, reduzindo a possibilidade destes dispositivos serem usados para fins

maliciosos. Entretanto, empresas especializadas em inovações tecnológicas afirmam que para o sucesso da IdC, tecnologias de gestão de identidades devem ser aprimoradas e adequadas às necessidades trazidas pela IdC. Dada a enorme quantidade de objetos ligados a Internet, é indispensável para o avanço e sucesso da IdC a existência de sistemas eficazes de gestão de identidades.

3.3.6 Cópias de Segurança

Diariamente os dispositivos da IdC lidam com dados sensíveis de utilizadores (e.g., empresas, negócios, bancos, governos, instituições não governamentais, etc.), e em grande escala. Contudo, a falta de um plano de contingência, facilita muitas vezes o trabalho de obtenção e comprometimento de informações sem nenhuma permissão ou dificuldade por parte de entidades maliciosas. Assim, de modo a precaverem-se estas situações, é importante que mecanismos como cópias de segurança de dados sejam considerados e colocados em funcionamento, tanto para a rede de suporte como para máquinas terminais. O uso de cópias de segurança permite uma maior confiabilidade na disponibilidade e capacidade de recuperação de sistemas, sendo normalmente a forma mais eficiente de restauro dos dados originais perdidos sem a necessidade de interferir no funcionamento normal da própria infraestrutura. A figura 3.4 ilustra a recolha de dados por parte dos dispositivos IdC na nuvem e posteriormente guardados em servidores que facilitam a sua gestão e manutenção.

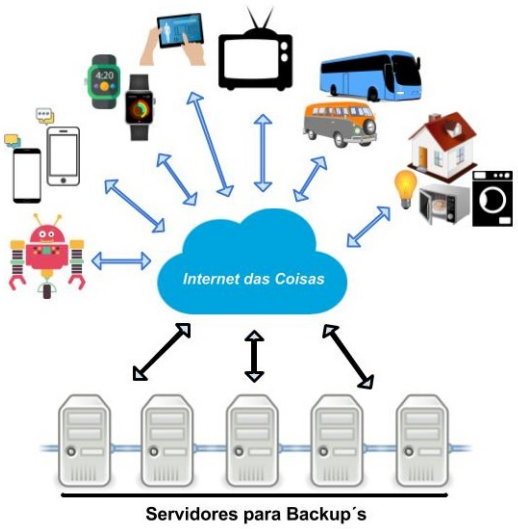

Figura 3.4: Representação da forma de gestão e cópia de segurança de dados da IdC recorrendo a servidores remotos.

As cópias de segurança são importantes e indispensáveis para as organizações que diariamente lidam com informações sensíveis, e os serviços na nuvem são amplamente usados neste contexto, sendo a redundância e a facilidade de integrar mecanismos de cópias de segurança uma das grandes bandeiras do paradigma da computação na nuvem. Contudo, As empresas usam bastante este recurso e serviços devido aos seus altos níveis de disponibilidade, degenerando em uma maior confiança por parte destes e fortalecimento da credibilidade da empresa. Outra vantagem que este paradigma apresenta é o acesso remoto. Quando combinado com a IdC, permite que pessoas e empresas possam aceder aos seus dados na nuvem em qualquer lugar e em qualquer hora com apenas a existência de Internet.

3.3.7 IDS, Honeypot e Antivírus

É hoje comum ouvirem-se noticias sobre ataques *hackers* em sistemas informáticos e dispositivos inteligentes, sendo que estes dispositivos apresentam crassas falhas de segurança, as quais são aproveitadas por intrusos para lançarem os seus ataques roubando informações valiosas e confidenciais ou usarem os dispositivos como trampolim para outros ataques. A falta de mecanismos de segurança e a não verificação da integridade do código de implementação dos sistemas embutidos nesses dispositivos estão na base da maior parte das vulnerabilidades. Implementações ou instalações de Sistemas Detetores de Intrusõess (IDSs), *honeypots* e o uso de *Antivírus* podem ajudar nestes casos, dado permitirem por vezes detetar, bloquear ou fornecer meios de análise a ameaças a um grande número de dispositivos, independentemente dos mecanismos de segurança embutidos nas Coisas.

Os IDSs são sistemas utilizados para detetar ações indevidas em infraestruturas de rede ou máquinas terminais. Permitem normalmente monitorizar as operações de uma rede, ou de um equipamento ligado a essa rede, informando os responsáveis quando encontrar alguma irregularidade (violação na política de segurança da rede e sinais de invasões). É um mecanismo de defesa muito utilizado hoje em dia, normalmente integrado em soluções de segurança complexos.

Os *honeypots* são sistemas usados no contexto da segurança de redes e sistemas para análise do tráfego e atividades maliciosas que exploram as vulnerabilidades amplamente conhecidas em ambientes da IdC. O principal objetivo destes sistemas é servir de engodo a atacantes, expondo algumas vulnerabilidades de propósito, enquanto se monitorizam e registam as atividades maliciosas e código nelas usadas, e posteriormente, extrapolar regras ou conclusões que permitam possivelmente evitar futuros ataques nestes ambientes [ATN17].

Outro software de segurança não menos importante, mas pouco utilizado, se não mesmo inexistente em dispositivos IdC, são os antivírus. Na verdade os antivírus são instâncias de IDSs para sistemas anfitriões. É muito comum instalar uma aplicação de antivírus em computadores com poder de processamento adequado (e.g., em computadores pessoais e servidores). No entanto, torna-se difícil a implementação desses mecanismos em ambientes IdC, sendo que alguns dispositivos nem um sistema operativo que os suporte possuem, pois o código todo está incorporado no *firmware*. Nesses casos, a escassa memória em sí pode inviabilizar a implementação ou integração de antivírus. Atualmente, empresas como a *BitDefender*, a *Kaspersky* e a *F-Secure* estão a investir em ambientes IdC e a procurar maneiras de incorporar, ainda que parcialmente, funcionalidades de antivírus e segurança em dispositivos inteligentes voltados para a IdC.

3.3.8 Atestação e Funções de *Hash*

A atestação está intrinsecamente interligada com o conceito de *Trusted Computer Base* (TCB), que é um conjunto de mecanismos de segurança que envolvem *(hardware, software e firmware)* e que são essenciais e indispensáveis para garantir que sistemas computacionais (eventualmente dispositivos IdC) sejam confiáveis, no sentido em que fazem exatamente aquilo que foram especificados para, não podendo a especificação ser alterada (e.g., maliciosamente) sem que isso seja detetado. O TCB é uma parte do sistema que se crê inviolável por desenho e construção, e que é responsável pela aplicação de políticas de segurança em todo o sistema, podendo incluir mecanismos de segurança como controlo de acesso, criptografia, e funcionalidades de *firewall* [LXW11]. A atestação e a inclusão de hardware e software dedicado a TCB está a tornar-se

mais comum à medida que o tempo avança. É frequente, por exemplo, as consolas de vídeo-jogos e os telemóveis topo de gama terem chips de criptografia dedicados, bem como uma parte do sistema operativo (um *realm*) com *Application Programming Interface* (API) limitada às funções de segurança que oferece. Se bem implementado, as funções executadas no TCB são consideradas corretas (e seguras) e imunes a infeções por código malicioso.

O TCB é eficaz na verificação e atestação contra a adulteração das entidades com acesso ao sistema. Muitas intrusões, nomeadamente as mais avançadas e persistentes (*Advanced Persistent Threats*) recorrem, em algum dos seus passos, à alteração do fluxo de execução do sistema operativo e programas através da injeção ou modificação de código. Uma vez que o código ou executáveis do dispositivo podem ser processados como qualquer outro tipo de dados, é possível efetuar testes de integridade com cadência predefinida e de modo a detetar adulterações. É claro que a atestação requer que se guarde o resumo inicial do sistema correto em local seguro (memória isolada, sem permissões de escrita), de modo a que este estado possa ser comparado com o do sistema em execução durante a atestação. A atestação que recorre a funções de *hash* é tipicamente considerada eficiente e encontrará aplicação na IdC.

Obviamente, a utilidade do TCB não se fica pela atestação. Podem-se ainda mencionar algumas funções básicas do TCB:

- Mediar todos os acessos;
- proteger o *software* e *firmware* contra modificações;
- Atestar as operações de entrada e saída;
- Garantir a credibilidade nos canais de comunicações;
- Fornecer outros recursos de proteção.

O TCB é um dos conceitos fundamentais para o objetivo da construção de uma IdC segura por construção, ainda que não seja realista pensar que sensores ou dispositivos de baixo custo tenham de o integrar. A sua integração deve ser alvo de reflexão aquando do desenho de sistemas, e a sua penetração deverá seguir o fluxo da disponibilidade de computação, de onde há mais para onde há menos. Na perspetiva da garantia de comunicações seguras e confiáveis, é necessário atestar a fiabilidade de execução dos dispositivos IdC de forma eficiente. Caso contrário, a grande quantidade de dados partilhados por estes dispositivos inteligentes pode trazer maior prejuízo para os utilizadores como para as empresas em geral [PK17].

3.3.9 Segurança Física

No contexto da IdC faz sentido falar na segurança física talvez ainda de forma mais incidente que para outros ambientes, dada a imersão inerente ao paradigma. A segurança física refere-se a todos os mecanismos que permitem que os dispositivos integrantes numa infraestrutura de rede estejam protegidos contra desastres locais, ambientais (terramotos, inundações, incêndios, etc.) e ainda contra extravio de equipamentos e acesso físico indevido. No entanto, é importante planear os métodos de segurança física para que os próprios dispositivos não estejam à mercê dos intrusos e que isso acarrete a perda de dados importantes e cruciais. Atualmente, a segurança física é um aspeto mais presente nas empresas que em qualquer um dos outros

ambientes referidos, onde a gestão de entrada e saída de pessoais e equipamentos se vem melhorando, por via da necessidade ao longo da história. Curiosamente, muitos dispositivos da IdC de hoje são também eles integrantes da segurança física de organizações e lares. Produtos como câmaras de segurança, fechaduras inteligentes, alarmes inteligentes, e carros inteligentes também desempenham papéis importantes na segurança física dos próprios bens ou pessoas, cativando o interesse de entidades maliciosas.

Por um lado, os fabricantes de dispositivos são culpados pela falta de segurança nos próprios dispositivos porque estão mais focados na margem de lucros das vendas, na facilidade de uso, limitando as funcionalidades e capacidades, deixando-os vulneráveis para os atacantes. Por outro lado, os utilizadores não ficam de fora dessa culpa, porque a maioria nem sequer se preocupa com aspetos básicos de segurança dos mesmos, mesmo a nível físico. A segurança física é de extrema importância na IdC porque esses dispositivos podem ser usados para controlar muito dos outros dispositivos inteligentes de forma remota e correm o risco de serem pirateados ou mesmo roubados por falta de um controlo adequado relativamente aos aspetos físico dos dispositivos. Por tanto, a falta de segurança física é apontado como um dos principais pontos fracos dos dispositivos inteligentes [MD16].

3.3.10 Políticas de Segurança

A fraca convergência na regulamentação dos dispositivos IdC e a má gestão dos mesmos leva a que muitos "buracos" de segurança sejam explorados e usados para fins maliciosos por entidades por vezes desconhecidas. O uso de políticas de segurança é um caminho viável para colmatar parte deste problema. Uma Política de Segurança é um conjunto de regras para garantir que todos os utilizadores dentro de um ambiente (neste caso do ecossistema IdC) atendam às prescrições relativas à segurança dos dados partilhados e na utilização dos equipamentos [Alq17], e especificam relações de confiança entre entidades, regras de proteção para qualquer recurso existente no sistema. Portanto, a utilização de uma política de segurança requer um estudo aprofundado do estado do ambiente em si, e dos intervenientes deste mesmo ambiente. Ao se lidar com tecnologias pouco seguras (como é o caso da IdC) um aspeto importante a ter em conta antes da implementação das políticas de segurança é analisar até que ponto os dispositivos e os próprios dados partilhados correm riscos significativos, e uma maneira adequada de identificar se esses riscos são endereçados é através da utilização de ferramentas de monitorização e relatórios.

Ambientes como a IdC necessitam de uma estrutura de segurança fiável e um conjunto de políticas de segurança auto adaptativas, que incluam tanto o sistema como um todo, como os dispositivos em particular. Repare-se que um ambiente IdC é potencialmente complexo em termos de número de equipamentos, dispersão geográfica, gestão e comunicações. No entanto, em muitos os casos, essas políticas não satisfazem os requisitos e os mecanismos de segurança implementados em ambientes IdC, e isso leva a outros problemas de segurança e a um sistema incompleto. Há uma resistência natural a formalizar medidas à cabeça, sendo muito comum formular resoluções após os problemas acontecerem, pelo que o tema das políticas de segurança é sensível. Para além disso, constata-se que essa resistência é agravada por certas mudanças ambientais, políticas ou até mesmo comerciais. Talvez o desenho de ferramentas que ajudem na elaboração semi-automática dessas políticas possa vir a fazer um contributo importante nesta área.

3.3.11 Comunicação Segura

O conjunto de tecnologias de comunicação existentes no mundo da IdC, como mencionado na subsecção 2.2.5 do capítulo 2, ilustram o vasto leque que os dispositivos inteligentes possuem de se comunicar com os demais objetos e computadores em vários cenários para IdC. Além disso, estes dispositivos possuem a capacidade de interagir diretamente com outras entidades ligadas à Internet, localizadas muito além do seu ambiente local. Na data de escrita desta dissertação e em alguns casos, as soluções disponíveis para garantir a segurança das comunicações não são ainda as melhores para a parte que mais perto está das coisas, especialmente quando se tratam de dispositivos com muito fraco poder de processamento, como por exemplo comunicações entre sensores e *Gateways*. Protocolos tipicamente utilizados na Internet (e.g., TLS) podem ser usados em partes da infraestrutura, mas nem sempre na sua totalidade, pela sua complexidade computacional, e as versões/adaptações mais leves desses protocolos não estão ainda suficientemente enraizados.

Para fornecer e garantir segurança nas comunicações nestes ambientes, é ainda necessário que tecnologias inovadoras e adequadas a dispositivos mais limitados sejam desenvolvidas. Em alguns casos, será interessante que essas tecnologias inter-operem com com as existentes. Os mecanismos leves de criptografia são, de resto, tópicos muito ativos nesta área do conhecimento.

3.3.12 Contas de Utilizador e Senhas

A ubiquidade de dispositivos inteligentes na sociedade deveria indicar a participação regular das pessoas no processo da segurança, nomeadamente em atos simples como o de autenticação pessoal em dispositivos detidos. As contas de utilizador e senhas ou palavras-passe (sistemas de *login*) são formas de autenticação e controlo de acesso muito populares nos dias de hoje, se não mesmo as mais usadas. No contexto dos dispositivos IdC ligados à Internet são provavelmente a mais comum medida de segurança que estes dispositivos usam contra possíveis ataques, senão a única. No entanto, na maior parte dos casos, as senhas fogem do padrão que é considerado seguro: são criadas com base em nomes, números, datas, lugares, sem nenhuma complexidade e sem relação com o tipo de conteúdo a partilhar pela Internet. Muitas credenciais de acesso nunca são mudadas ao longo da vida de um dispositivo da IdC, sendo por vezes públicas as palavras-passe que vêem de fábrica em determinados equipamentos. Atitudes assim, permitem que os dispositivos da IdC estejam vulneráveis a vários ataques que visam assumir o controlo dos mesmos, e usá-los para fins maliciosos.

Não admira portanto que este tipo específico de informação seja de especial interesse para atacantes. Alguns dos ataques de *phishing* mais comuns (e que recorrem, e.g., ao uso de um sítio *web* falso, são usados para capturar precisamente senhas e outras informações pessoais. É também frequente verem-se grandes bases de dados de credenciais de acesso serem vazadas para a Internet ou serem vendidas *online*. Assim, é clara a necessidade de melhorar as técnicas de autenticação, numa primeira fase talvez através de técnicas de *autenticação em vários fatores*, para depois evoluir para técnicas baseadas em criptografia moderna mais seguras.

3.4 Mapeando os Requisitos de Segurança

Os requisitos e os mecanismos de segurança abordados no presente capítulo representam parte dos meios pelos quais a IdC se pode vir a tornar segura por desenho. O processo da garantia da segurança por construção não é absoluto, pelo que apenas se podem fazer afirmações relativas e modestas a este aspeto. Garantir segurança em ambientes inteligente não é uma tarefa fácil e passa por um estudo exaustivo e consequentemente implementação de mecanismos viáveis para cada ambiente específico. Um dos objetivos principais desta dissertação era fazer um mapeamento inicial entre os requisitos e mecanismos de segurança para a IdC, pois a maioria dos requisitos de segurança são aplicados as tecnologias de informação, no entanto, poucos são os que se adaptam especificamente a este paradigma. A ideia não é oferecer um mapeamento final, mas contribuir para o caminho.

A tabela 3.1 apresenta um mapeamento entre os requisitos e os mecanismos de segurança discutidos na dissertação até aqui. De salientar que este mapeamento pode ser válido para outros ambientes (que não o da IdC), embora tivesse sido feito especificamente a pensar nestes ambientes. Na tabela, foi usado o símbolo ✓ para identificar quais os requisitos parcial ou totalmente preenchidos por determinado mecanismo de segurança, e um espaço em branco para identificar os requisitos que não são preenchidos por determinado mecanismo, respetivamente. Adicionalmente, abreviaram-se os nomes dos requisitos de segurança de acordo com a seguinte chave: *Priv = Privacidade, Conf = Confidencialidade, Integ = Integridade, Dispo = Disponibilidade, Aut = Autenticidade, Auto = Autorização, Auten = Autenticação, N-Repu = Não-Repúdio, Confo = Conformidade, Confi = Confiança, Audi = Auditoria*.

Tabela 3.1: Mapeamento entre requisitos e mecanismos de segurança para a IdC.

Mecanismos de Segurança	Requisitos de Segurança										
	Priv	Conf	Integ	Dispo	Aut	Auto	Auten	N-Repu	Confo	Confi	Audi
Atualizações Autenticadas	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Secure Booting	✓	✓	✓	✓	✓	✓	✓			✓	
Firewall	✓	✓	✓	✓	✓	✓			✓	✓	✓
Criptografia	✓	✓	✓	✓	✓	✓	✓			✓	
Assinaturas Digitais	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓
Certificados Digitais	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Funções de Hash	✓	✓	✓	✓		✓		✓	✓	✓	
Controlo de Acesso	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓
Cópias de Segurança	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Segurança Física	✓		✓	✓	✓	✓			✓		
Antivírus	✓		✓	✓	✓	✓	✓		✓	✓	✓
IDS	✓		✓	✓	✓	✓			✓	✓	✓
Honeypot	✓		✓	✓	✓	✓					
Comunicação Segura	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Contas de Utilizador e Senhas	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓
Gestão de Identidades	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Como se pode concluir da análise da tabela, é possível fazer um mapeamento entre todos os requisitos e os mecanismos existentes. Por vezes, um requisito pode ser preenchido por mais do que uma tecnologia ou mecanismo, mas deve ser referido que a aplicação de um único mecanismo pode ser já complicado em dispositivos com limitações, quanto mais a combinação de vários. Requisitos como Confidencialidade, Integridade e Disponibilidade representam as propriedades mais básicas que um sistema ou dispositivo deve implementar para assegurar que tanto a funcionalidade dos dispositivos como a privacidade dos dados não tenham destinos inapropriados. Os demais requisitos representam a necessidade que o mundo da IdC tem de expandir a segurança em todos os seus níveis de operação.

Um outro aspeto importante a realçar é que, embora vários dispositivos inteligentes não suportem alguns dos mecanismos acima citados por enquanto, existem previsões (como aquela que é feita em [Nor16]) que apontam para que uma maior fatia desses objetos poderão estar aptos a suportarem mecanismos mais fortes num futuro próximo, com base também no que aconteceu em ambientes convencionais de rede onde os dispositivos possuem maiores capacidades em termos de memória, processamento e energia.

3.5 Conclusão

À medida que a IdC vai se expandido, a diversidade e a complexidade na recolha e tratamento dos dados partilhados nesta rede aumenta significativamente. Os ambientes inteligentes construídos através de tecnologias da IdC são vulneráveis a ataques que visam roubar informações valiosas e confidenciais, controlar dispositivos, e interromper serviços. A segurança da IdC não é totalmente compreendida quando comparada com outros ambientes mais sofisticados e evoluídos, por estar numa fase embrionária de desenvolvimento. No entanto, as questões de segurança devem ser planeadas e implementadas desde o início da criação de ambientes inteligentes (e.g., residencial, hospitalar, industrial, comercial, empresarial) se estes problemas são para serem resolvidos na sua raiz.

Neste capítulo, foram apresentados os requisitos e mecanismos de segurança específicos para a IdC, considerando a importância exercida pela privacidade e, relevando a implementação destes requisitos e mecanismos para os dispositivos, assim como para os sistemas da IdC. Foi também incluído um mapeamento genérico entre os requisitos e mecanismos de segurança, após se discutirem ambos com mais detalhe. A lista de requisitos não é considerada completa, mas todos os requisitos ou propriedades de segurança encontrados e mencionadas neste capítulo são indispensáveis para concederem aos utilizadores o total controlo dos seus dados, e estes, disponibilizarem apenas o que convém.

De realçar ainda que a maioria dos modernos dispositivos e sistemas de hoje em dia são complexos, por vezes demasiado recetivos a *inputs* do exterior e disponíveis, e portanto bastante vulneráveis a ameaças. Garantir a segurança nestes dispositivos é considerada uma tarefa difícil por causa das restrições impostas em termos de energia, processamento e memória. Ficou óbvio ser necessário fazer uso urgente de mecanismos seguros e eficientes capazes de protegerem estes dispositivos contra ameaças e ataques realizados por entidades maliciosas. Estas são as etapas para tornar os ambientes inteligentes mais seguros e ajudarem a criar um vínculo inquebrável entre a IdC, a sociedade, as empresas, e as indústrias.

Capítulo 4

Testes em Plataformas da Internet das Coisas e Arquitetura para o Mapeamento

4.1 Introdução

À data de escrita da dissertação existem já inúmeros dispositivos e plataformas da IdC [SSC+18], com um enorme leque de capacidades computacionais e de comunicação. Há equipamentos com especificações semelhantes a computadores pessoais (e.g., memória *Random Access Memory* (RAM) superior a 1 GB e disco rígido dedicado), tal como há equipamentos bastante limitados e com sistemas operativos dedicados. Já aqui foi dado a entender que a utilização direta de mecanismos e tecnologias de segurança existentes nestes dispositivos ou plataformas pode não ser possível, mas que interessa fazer pela análise empírica da performance dessas tecnologias, que de resto é parte dos objetivos principais desta dissertação. O presente capítulo apresenta vários testes realizados para um conjunto de implementações de algoritmos criptográficos populares (foram usadas as implementações da biblioteca *OpenSSL*[1]) e para dois recursos de segurança do Linux (firewall e Snort). Os testes foram realizados numa plataforma existente para a IdC e num computador pessoal para comparação, e no final do capítulo são apresentados os resultados encontrados e as devidas conclusões. A secção 4.2 apresenta a descrição dos testes realizados em diferentes plataformas da IdC, a secção 4.3 apresenta os resultados obtidos dos testes realizados, a secção 4.4 apresenta a simulação de ataques dirigidas ao Raspberry Pi, testando os mecanismos de segurança nomeadamente o *iptables e o snort* e a secção 4.5 idealiza a arquitetura de uma ferramenta de segurança como prova de conceito utilizando os requisitos e os mecanismos de segurança apresentados nesta dissertação.

4.2 Descrição de Testes com Implementações OpenSSL

A presente secção descreve o método usado para a realização dos testes no dispositivos da IdC e no computador pessoal. Os algoritmos de cifra estudados durante estes testes foram os seguintes:

- o AES (um algoritmo de cifra de chave simétrica por blocos, norma internacional recomendada para todas as utilizações de cifra de chave simétrica na data de escrita da dissertação), nos modos de operação *Electronic Code Book* (ECB), *Cipher Block Chaining* (CBC), *Counter Mode* (CTR), *Output Feedback* (OFB), *Cipher Feedback* (CFB);
- o 3DES (um algoritmo de cifra de chave simétrica por blocos, ainda utilizada em algumas aplicações criptográficas), nos modos de operação CBC, OFB, CFB;
- o DES (um algoritmo de cifra de chave simétrica por blocos obsoleto mas importante por motivos históricos), nos modos de operação ECB, CBC, OFB, CFB;

[1] https://www.openssl.org/

- o RC4 (um algoritmo de cifra de chave simétrica contínua obsoleto, mas importante por motivos históricos e ainda usado em alguns casos); e
- o RSA (um algoritmo de cifra de chave pública mais utilizado à data de escrita da dissertação).

De salientar que estes algoritmos estão implementados na biblioteca *OpenSSL* e a maior parte deles são atualmente os mais utilizados no desenvolvimento de aplicações informáticas, mesmo a nível militar, seja para a *web*, servidores ou para dispositivos inteligentes da IdC. No entanto, muitos destes algoritmos não se adaptam com facilidade aos dispositivos com restrições de memória, processamento e consumo de energia, sendo por isso interessante comparar o comportamento destes algoritmos numa plataforma computacional tipicamente usada na IdC com o comportamento num computador normal. Assim sendo, a subsecção 4.2.1 apresenta de forma abrangente os algoritmos usados com os seus modos de cifra, e a 4.2.2 descreve os métodos usados para realização dos testes.

4.2.1 Algoritmos de Cifra

DES e 3DES: São algoritmos de cifra por blocos de tamanho fixo e de chave simétrica. O DES possui uma chave de 56 bits (7 bytes) e o tamanho do bloco é de 64 bits (8 bytes), enquanto o 3DES opera com chaves de 168 bits (21 bytes) e o tamanho do bloco é de 64 bits (8 bytes). O DES é atualmente considerado inseguro por possuir um tamanho de chave relativamente pequeno e por já haver casos de quebra da cifra [MPT12]. A utilização do DES é desaconselhada, mas foi aqui considerado pelo seu peso histórico (primeiro algoritmo de chave simétrica com norma internacional) e por ser o bloco básico de construção do 3DES, que combina a cifra-decifra-cifra com 3 chaves para obter o triplo da força criptográfica do DES. Ambas as cifras operam num esquema de rondas, e utilizam redes de Feistel no seu processo de cifra e decifra. Ambas as cifras foram utilizadas nos testes, operando nos modos de cifra: ECB, CBC, OFB, e CFB. *AES:* é um algoritmo de chave simétrica por blocos. O AES constitui atualmente a norma internacional para cifras de chave simétricas segura, sucedendo ao DES. A norma AES define que podem ser usados até três tamanhos de chaves diferentes, nomeadamente 128, 192 e 256 bits, que correspondem a diferentes níveis de segurança (longevidade temporal) oferecidas pela cifra. Independentemente do tamanho da chave, o tamanho do bloco sobre o qual a AES opera é sempre 128 bits [MPT12]. É possível operar a AES nos vários modos de cifra para cada tamanho de chave: e.g., ECB, CBC, CTR, OFB, CFB. Todos os modos enunciados neste parágrafo foram testados no contexto deste trabalho para o AES. Considera-se que, quando usada corretamente, a cifra AES é sinónimo de segurança e eficiência de desempenho, tornando-se uma excelente candidata para soluções de cifra de chave simétrica para os dispositivos e sistemas IdC.

RC4: é um algoritmo de cifra de fluxo, também designada por cifra de chave simétrica contínua. A primitiva RC4 foi um gerador de números pseudo-aleatórios criptograficamente seguro (foram identificadas chaves fracas e descobertas fragilidades) desenvolvido por Ron Rivest em 1987, e tem sido utilizada em vários ambientes e protocolos, sobretudo pela sua elevada eficiência computacional. O RC4 foi muito utilizado em padrões de segurança como o TLS, *Wired Equivalent Privacy* (WEP) e *Wi-Fi Protected Access* (WPA) [LF07] até 2015. O RC4 utiliza chaves de 40 a 128 bits, respetivamente. Foi considerado no contexto deste trabalho por ter um peso histórico considerável e como forma de analisar a performance em comparação com os demais algoritmos abordados neste artigo.

RSA: é um algoritmo criptográfico de chave pública ideal para cifrar e decifrar pequenas quantidades de dados (cadeias de *bits* menores que um dos parâmetros da chave pública conhecido como módulo). Foi criado por Ronald Rivest, Adi Shamir e Leonard Adleman em 1978. É considerado um dos maiores avanços da criptografia de chave pública [RSA78], sendo ideal para para cifrar (e trocar) segredos criptográficos e chaves de cifra simétricas, bem como para assinar valores de *hash* (tipicamente menores que o módulo). O RSA elabora na Teoria dos Números e a sua segurança deve-se sobretudo ao problema matemático intratável da fatorização de um número composto extremamente grande em primos. A cifra e decifra, na sua forma mais direta (sem preenchimento) corresponde a uma operação de exponenciação modular, simples de entender e programar (às quais também se devem a sua popularidade). Faz uso de funções de sentido único com alçapão, que não podem inverter em tempo útil a não ser que se saiba um segredo (a chave privada). Para cifrar uma mensagem $M \in \mathbb{Z}_n$, usa-se a chave pública pk constituída por dois números (N, e), e calcula-se $C \leftarrow M^e \pmod N$. Para decifrar o criptograma, usa-se a chave privada sk no cálculo $M \leftarrow C^d \pmod N$, onde M é a mensagem, C é o criptograma, e e d são números inteiros e N é o resultado da multiplicação de dois números primos considerados grandes ($\geq 2^{1024}$). De salientar ainda que os tamanhos de chaves apresentados acima foram utilizados neste trabalho para assinar e verificar ficheiros com tamanhos de 100MB e 1 GB e 2 GB utilizando a função de SHA256.

Funções de hashing: são funções criptográficas que geram resumos (de saída) de tamanhos fixos (normalmente entre 128 a 512 bits) independentemente do tamanho da mensagem ou arquivo (de entrada). O resumo é considerado como sendo o *hash* da mensagem (ou o que quer que seja a entrada). Para ter utilidade criptográfica e ser considerada como segura, a função de *hashing* deve ter as seguintes propriedades principais:

- Unidirecional: dado um *hash* de uma mensagem *h(M)*, deve ser impossível computacionalmente encontrar *M* a partir do mesmo hash.
- Difusão: dada uma mensagem *M*, deve ser impossível alterar o seu conteúdo sem alterar o hash *h(M)*.
- Colisão: dada uma mensagem *M* deve ser impossível encontrar outra mensagem *M'* tal que *h(M) = h(M')*.

As funções de *hash* bastante usadas atualmente e que foram usadas nesta dissertação, são MD5, o SHA1, SHA256 e o SHA512. O MD5 é uma função de 128 bits. Nos dias de hoje, é amplamente usada no mundo do software como forma de garantir integridade de arquivos ou mensagens. No entanto, é desaconselhado o seu uso para fins criptográficos por serem conhecidas diversas vulnerabilidades (e casos famosos em que elas foram exploradas). As funções SHA1, SHA256 e o SHA512, embora sendo da mesma família, são estruturalmente diferentes. Geram resumos de 160, 256 e 512 bits, respetivamente. Consideradas mais seguras que o MD5, o SHA1 e o SHA256 são largamente usadas em aplicações, funcionalidades e protocolos criptográficos (e.g., TLS) e são ainda as funções padrão para as assinaturas digitais.

4.2.2 Descrição do Método Usado

Os teste apresentados nesta dissertação, permitiram substanciar até certo ponto conclusões acerca dos algoritmos de cifra com maior potencial de adaptação ao mundo da IdC (e.g., dispo-

sitivos e sistemas) com nível de recursos semelhantes aos do nosso dispositivo de teste, através da eficiência na resposta dada pela utilização dos seus métodos de cifrar e decifrar. Os testes basearam-se na obtenção do tempo gasto, em segundos, e do consumo de memória em *Kilobytes*. Além dos cálculos referentes ao tempo e a memória isolados, foram depois obtidos valores estatísticos como a variância e o desvio padrão.

Para efetuar os testes foram utilizadas as implementações dos algoritmos definidas na biblioteca `OpenSSL`, versão 1.1.0.2g, com chaves de cifra e vetores de inicialização aleatórios, criadas a partir do comando `openssl rand -hex <tamanho em byte>`. Cada teste consistia na cifra de um ficheiro e nas medições dos parâmetros definidos de seguida. Para medição de tempo e consumo de memória, foi utilizada a ferramenta `time`, com os parâmetros `e` e `M`, respetivamente. A variância e desvio padrão foram obtidos a partir das fórmulas $s^2 = \sum \frac{(x_i - \bar{x})^2}{n-1}$ e $s = \sqrt{\sum \frac{(x_i - \bar{x})^2}{n-1}}$, respetivamente, onde x_i denotam as várias medições e $\bar{x}$ denota a média aritmética dos valores obtidos.

Os testes foram efetuados num *Raspberry Pi 3 (Cortex A53 Quad Core, ARM Cortex, 1.2 GHz, com 16 GB de armazenamento, memória de 1 GB com o sistema operativo Ubuntu MATE 16.04.2)* e num computador *ACER Aspire ES 15(AMD Quad-Core, A5-5000 de 1,5 GHz, HDD de 1 TB, memória de 4 GB DDR3 e com o sistema operativo Ubuntu 18.04)*. Foram realizadas um total de 100 repetições para cada ficheiro com tamanhos de 100 MB, 1 GB e 2 GB, tanto para o Raspberry Pi 3 como para o computador. Estes ficheiros foram criados usando o comando `dd if=/dev/zero of=<descrição> bs=<valor> count=<tamanho em MB>`, onde o tamanho fixo do ficheiro é obtido através da multiplicação de `bs` e `count`. As chaves privadas e públicas para a assinatura e verificação dos ficheiros usados na cifra RSA foram obtidas a partir dos comandos `openssl genrsa -out <nome_do_ficheiro> <tamanho_da_chave_em_bits> 1> /dev/null 2> /dev/stdout` e `openssl rsa -in <chave_privada> -pubout -out <chave_publica> 1> /dev/null 2> /dev/stdout`, respetivamente.

Listing 1 Trecho de código para a obtenção do tempo gasto de cifra para ficheiro de 1 GB.

```
#!/bin/sh
somaT=0
somaM=0
varT=0;
varM=0;
for i in {1..100}
do
iv=$(openssl rand -hex 16)
chave=$(openssl rand -hex 16)
tempo=$((/usr/bin/time -f '%e-%M' openssl enc -aes-128-cbc -K $chave -in 1GB -out ENC -iv $iv 1> /dev/null) 2>&1)
tem=${tempo:0:2}
mem=${tempo:3}
somaT=$(echo $somaT+$tem | bc -l)
somaM=$(echo $somaM+$mem | bc -l)
arrT[$i]=$tem
arrM[$i]=$mem
echo $i-$tempo
done
echo "--------------------------------"
mediaT=`echo "scale=2; $somaT/100" | bc -l`
mediaM=`echo "scale=2; $somaM/100" | bc -l`
echo "Média do tempo: " $mediaT
echo "Média da Memória: " $mediaM

for j in {1..100}
do
varT=$(echo "scale=2; $varT + ($mediaT-arrT[$j])*($mediaT-arrT[$j])/99" | bc)
varM=$(echo "scale=2; $varM + ($mediaM-arrM[$j])*($mediaM-arrM[$j])/99" | bc)
done
desvioT=$(echo "sqrt($varT)" | bc)
desvioM=$(echo "sqrt($varM)" | bc)
```

Os trechos de código 1 e 2 apresentam os `scripts` usados para a realização dos testes aos algoritmos criptográficos acima descritos. De salientar ainda que, em função dos testes realizados, estes `scripts` foram adaptados para as várias cifras e tamanhos de ficheiros em análise. Ainda de mencionar que foi criada uma ferramenta em JAVA que comprovava os resultados obtidos a partir destes `scripts`com os obtidos no programa assim implementado.

Listing 2 Trecho de código para a obtenção do tempo gasto na assinatura do ficheiro de 1 GB.

```
somaT=0
somaM=0
varT=0;
varM=0;
for i in {1..100}
do
openssl genrsa -out chave 1024 1> /dev/null 2> /dev/stdout
openssl rsa -in priv.key -pubout -out pub.key 1> /dev/null 2> /dev/stdout
tempo=$((/usr/bin/time -f '%e-%M' openssl dgst -sha256 -sign priv.key -in 1GB -out sign 1GB 1> /dev/null) 2>&1)
tem=${tempo:0:2}
mem=${tempo:3}
somaT=$(echo $somaT+$tem | bc -l)
somaM=$(echo $somaM+$mem | bc -l)
arrT[$i]=$tem
arrM[$i]=$mem
echo $i-$tempo
done
echo "--------------------------------"
mediaT=`echo "scale=2; $somaT/100" | bc -l`
mediaM=`echo "scale=2; $somaM/100" | bc -l`
echo "Média do tempo: " $mediaT
echo "Média da Memória: " $mediaM

for j in {1..100}
do
varT=$(echo "scale=2; $varT + ($mediaT-arrT[$j])*($mediaT-arrT[$j])/99" | bc)
varM=$(echo "scale=2; $varM + ($mediaM-arrM[$j])*($mediaM-arrM[$j])/99" | bc)
done
desvioT=$(echo "sqrt($varT)" | bc)
desvioM=$(echo "sqrt($varM)" | bc)
```

4.3 Resultados dos Testes ao OpenSSL

As tabelas incluídas nesta secção apresentam os resultados da performance relativamente ao tempo gasto pelos algoritmos DES, 3DES, RC4 e AES para cifrar e decifrar um ficheiro de 100 MB e 1 GB, para o Raspberry Pi 3 e o Computador Pessoal. É possível observar que o algoritmo AES, em dispositivos com poder computacional razoável, como o Raspberry Pi 3 utilizado, se mostra o mais eficiente, tanto para ficheiros de menor como de maior tamanho. A cifra 3DES, por sua vez, apresenta um desempenho bastante inferior, enquanto que, em ficheiros de maior dimensão, a cifra RC4 se aproxima do AES. No entanto, e como explicitado na secção anterior, a cifra AES apresenta uma maior segurança, sendo, por isso, a melhor escolha entre o grupo analisado. Quanto ao tamanho das chaves utilizadas, não existem diferenças significativas (a maior variação é de cerca de 12%), não havendo por isso benefício na utilização de uma chave de tamanho menor. Quanto ao caso do algoritmo de assinatura RSA, os resultados obtidos, em ambos os dispositivos, demonstram uma performance em linha com o esperado, sendo cerca de três vezes mais rápido no seu processo do que o AES no seu processo de cifra (note-se que a assinatura digital é feita para resumos de ficheiros, não para o ficheiro). Quando comparando os dois dispositivos, o computador, como esperado, consegue efetuar as mesmas operações em cerca de 15 a 25% do tempo.

As tabelas contidas nos Anexo A (Tabelas A.1 a A.12) apresentam a totalidade dos resultados relativamente ao tempo gasto em segundos e ao consumo de memória em *Kilobytes* pelos processos

dos diferentes algoritmos e para os dois dispositivos, para ficheiros de 100 MB, 1 GB e 2 GB. Essas tabelas contêm os somatórios do tempo gasto e do consumo de memória de cada operação relativa aos algoritmos criptográficos nas colunas $\sum_{i=1}^{100} t_i$ e $\sum_{i=1}^{100} m_i$, respetivamente, $\bar{x}$ representa a média do somatório, s^2 representa a variância, e s representa o desvio padrão. É de chamar a atenção para a diferença de memória utilizada quando comparados os dispositivos, com o Raspberry Pi 3 a consumir cerca de 6% de memória em algumas operações. Estes consumos, na ordem dos 2500 KB, demonstram que estes algoritmos têm requisitos baixos de memória. É também possível observar que o tamanho do ficheiro não tem influência nesta métrica, sendo os valores obtidos constantes entre estes. Estes resultados são transversais aos diferentes algoritmos, não existindo, por isso, vantagem entre estes. O tamanho das chaves, mais uma vez, não tem influência nos resultados obtidos.

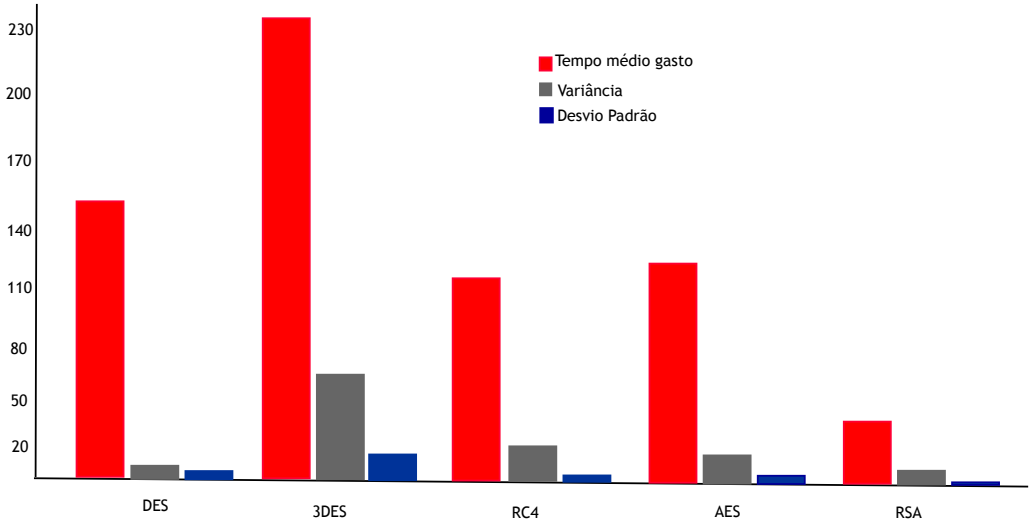

Figura 4.1: Comparação de performance (tempo médio gasto em segundos) entre as cifras e mecanismos de assinatura digital DES, 3DES, RC4, AES e RSA, para ficheiro de 1 GB no Raspberry Pi 3.

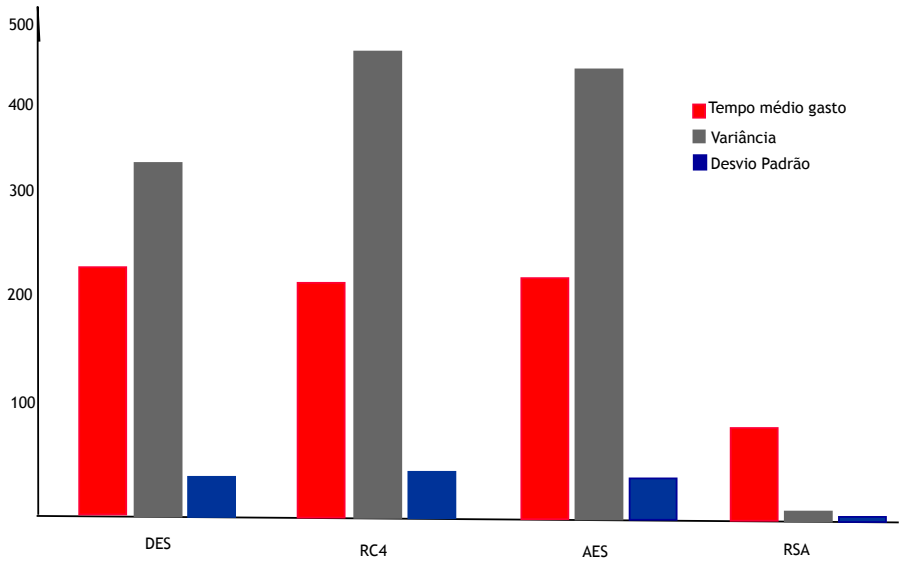

Figura 4.2: Comparação de performance (tempo médio gasto em segundos) entre as cifras e mecanismos de assinaturas digital DES, RC4, AES e RSA, para ficheiro de 2 GB no Raspberry Pi 3.

Uma nota importante que foi constatada durante a realização dos testes ao *OpenSSL* é que os resultados da cifra 3DES usando o ficheiro de 2 GB não foram satisfatórios. Ou seja, o Raspberry Pi 3 não suportou as operações realizadas pela cifra e como consequência danificou por duas vezes o recurso de armazenamento (cartão de memória). Com isso, conseguimos perceber que para ficheiros acima de 1 GB, o 3DES no Raspberry Pi 3 apresenta um comportamento bastante diferente relativamente às demais cifras.

Tabela 4.1: Resultados referente ao tempo gasto (em segundos) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 1 GB para o Computador

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
1 GB	AES	ECB	128	2467,26	24,67	35,00	5,92	1601,38	16,01	15,25	3,91
			192	2664,07	26,64	11,28	3,36	1229,39	12,29	1,72	1,31
			256	1631,16	16,31	5,46	2,34	1436,13	14,36	1,12	1,06
		CBC	128	2851,55	28,52	0,53	0,73	2350,28	23,50	1,71	1,31
			192	3302,17	33,02	3,10	1,76	3436,39	34,36	5,96	2,44
			256	3129,28	31,29	19,46	4,41	2838,06	28,38	88,36	9,40
		CTR	128	2810,13	28,10	0,99	1,00	1555,23	15,55	32,07	5,66
			192	3194,07	31,94	2,57	1,60	3291,67	32,92	5,68	2,38
			256	3252,86	32,53	16,17	4,02	3144,05	31,44	91,82	9,58
		OFB	128	2891,66	28,92	0,64	0,80	2819,59	28,20	76,95	8,77
			192	3585,15	35,85	3,99	2,00	3562,08	35,62	6,59	2,57
			256	3131,90	31,32	0,84	0,92	2473,37	24,73	1,01	1,01
		CFB	128	3137,49	31,37	0,78	0,88	2706,48	27,06	3,22	1,80
			192	3106,87	31,07	5,31	2,30	2480,89	24,81	2,96	1,72
			256	3141,68	31,42	2,90	1,70	2891,57	28,92	2,45	1,57
	DES	ECB	56	6037,24	60,37	11,60	3,41	5880,85	58,81	8,41	2,90
		CBC	56	5920,07	59,20	9,63	3,10	5777,89	57,78	7,70	2,78
		OFB	56	6338,13	63,38	8,90	2,98	6197,55	61,98	10,30	3,21
		CFB	56	6561,84	65,62	7,02	2,65	6434,43	64,34	13,04	3,61
	3DES	CBC	168	12854,02	128,54	4,47	2,11	12935,03	129,35	9,32	3,05
		OFB	168	13243,61	132,44	4,11	2,03	13306,85	133,07	8,08	2,84
		CFB	168	13512,32	135,12	8,39	2,90	13557,25	135,57	8,51	2,92
	RC4		128	1664,55	16,65	27,94	5,29	1662,27	16,62	35,34	5,94
	RSA		1024	1303,49	13,03	0,02	0,13	1302,27	13,02	0,00	0,02
			2048	1303,30	13,03	0,00	0,03	1301,89	13,02	0,00	0,02
			4096	1306,94	13,07	0,00	0,02	1302,21	13,02	0,00	0,02
			8192	1330,30	13,30	0,00	0,02	1302,29	13,02	0,00	0,03
			16384	1507,18	15,07	0,00	0,03	1303,03	13,03	0,00	0,02
	HASH	MD5	128	414,79	4,15	0,90	0,95				
		SHA	160	622,77	6,23	0,00	0,02				
			256	1297,67	12,98	0,00	0,02				
			512	1052,05	10,52	0,00	0,03				

Tabela 4.2: Resultados referente ao consumo de memória (em kilobytes) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 1 GB para o Computador Pessoal.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s
1 GB	AES	ECB	128	431052,00	4310,52	8591,08	92,69	431196,00	4311,96	10565,65	102,79
			192	437828,00	4378,28	7173,90	84,70	437580,00	4375,80	5456,61	73,87
			256	434024,00	4340,24	8010,29	89,50	433036,00	4330,36	5771,34	75,97
		CBC	128	441852,00	4418,52	5265,02	72,56	442504,00	4425,04	5237,53	72,37
			192	441740,00	4417,40	5610,14	74,90	442664,00	4426,64	5886,78	76,73
			256	440492,00	4404,92	5950,34	77,14	442068,00	4420,68	7712,99	87,82
		CTR	128	441504,00	4415,04	5380,89	73,35	441060,00	4410,60	6750,51	82,16
			192	443916,00	4439,16	5351,21	73,15	443376,00	4433,76	5355,74	73,18
			256	442508,00	4425,08	5419,91	73,62	443036,00	4430,36	4832,35	69,52
		OFB	128	445072,00	4450,72	3714,87	60,95	438284,00	4382,84	4577,39	67,66
			192	437720,00	4377,20	3969,13	63,00	438556,00	4385,56	5094,75	71,38
			256	437916,00	4379,16	4534,40	67,34	438832,00	4388,32	4827,05	69,48
		CFB	128	439284,00	4392,84	4273,23	65,37	440196,00	4401,96	3234,42	56,87
			192	434580,00	4345,80	10629,62	103,10	435760,00	4357,60	6902,63	83,08
			256	438284,00	4382,84	4959,77	70,43	438084,00	4380,84	5279,77	72,66
	DES	ECB	56	438156,00	4381,56	10867,04	104,25	436796,00	4367,96	11994,67	109,52
		CBC	56	435896,00	4358,96	10885,37	104,33	436940,00	4369,40	11511,07	107,29
		OFB	56	437808,00	4378,08	12692,52	112,66	437584,00	4375,84	10022,12	100,11
		CFB	56	437472,00	4374,72	12198,10	110,45	437504,00	4375,04	12070,83	109,87
	3DES	CBC	168	434544,00	4345,44	5169,30	71,90	432228,00	4322,28	4765,50	69,03
		OFB	168	432136,00	4321,36	5542,21	74,45	432608,00	4326,08	4357,00	66,01
		CFB	168	434276,00	4342,76	4776,31	69,11	432784,00	4327,84	6060,90	77,85
	RC4		128	418916,00	4189,16	17623,37	132,75	419824,00	4198,24	14847,46	121,85
	RSA		1024	448668,00	4486,68	3609,23	60,08	455512,00	4555,12	3725,12	61,03
			2048	449272,00	4492,72	3668,49	60,57	455480,00	4554,80	3952,65	62,87
			4096	451808,00	4518,08	4402,90	66,35	454284,00	4542,84	4259,97	65,27
			8192	454056,00	4540,56	4775,76	69,11	456440,00	4564,40	3121,29	55,87
			16384	461620,00	4616,20	3775,15	61,44	458268,00	4582,68	3234,28	56,87
	HASH	MD5	128	431464,00	4314,64	3835,22	61,93				
		SHA	160	434240,00	4342,40	2592,97	50,92				
			256	434404,00	4344,04	3813,01	61,75				
			512	433928,00	4339,28	5736,53	75,74				

Tabela 4.3: Resultados referente ao tempo gasto (em segundos) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 1 GB para o Raspberry Pi.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
1 GB	AES	ECB	128	13268,68	132,69	18,76	4,33	13565,16	135,65	15,11	3,89
			192	13218,52	132,19	10,37	3,22	13589,07	135,89	23,51	4,85
			256	13311,33	133,11	31,47	5,61	13584,00	135,84	16,83	4,10
		CBC	128	13238,24	132,38	12,62	3,55	13709,44	137,09	19,36	4,40
			192	13339,49	133,39	15,71	3,96	13695,60	136,96	15,50	3,94
			256	13352,38	133,52	12,96	3,60	13622,41	136,22	13,15	3,63
		CTR	128	13156,17	131,56	6,59	2,57	13402,20	134,02	13,15	3,63
			192	13175,23	131,75	7,95	2,82	13379,24	133,79	9,77	3,13
			256	13198,99	131,99	18,33	4,28	13431,21	134,31	10,73	3,28
		OFB	128	13695,29	136,95	25,95	5,09	13807,77	138,08	27,38	5,23
			192	13699,71	137,00	23,56	4,85	13870,51	138,71	21,69	4,66
			256	13704,05	137,04	31,37	5,60	13918,14	139,18	35,73	5,98
		CFB	128	13755,33	137,55	34,97	5,91	13834,72	138,35	19,68	4,44
			192	13618,88	136,19	14,80	3,85	13904,42	139,04	16,36	4,04
			256	13688,96	136,89	14,78	3,84	13987,86	139,88	27,54	5,25
	DES	ECB	56	14605,58	146,06	6,68	2,58	14600,53	146,01	15,14	3,89
		CBC	56	14036,01	140,36	8,70	2,95	14342,30	143,42	8,87	2,98
		OFB	56	14879,14	148,79	8,38	2,90	15182,48	151,82	13,40	3,66
		CFB	56	14847,56	148,48	12,51	3,54	15048,66	150,49	10,48	3,24
	3DES	CBC	168	21655,70	216,56	67,20	8,20	22569,88	225,70	5,58	2,36
		OFB	168	22409,19	224,09	73,37	8,57	23176,36	231,76	5,24	2,29
		CFB	168	22393,54	223,94	79,43	8,91	23356,21	233,56	9,05	3,01
	RC4		128	12671,29	126,71	35,05	5,92	13308,27	133,08	6,38	2,53
	RSA		1024	4374,23	43,74	5,67	2,38	4365,44	43,65	4,33	2,08
			2048	4366,32	43,66	4,38	2,09	4363,06	43,63	4,37	2,09
			4096	4398,95	43,99	7,01	2,65	4378,82	43,79	5,24	2,29
			8192	4460,85	44,61	4,22	2,06	4368,00	43,68	4,24	2,06
			16384	5056,16	50,56	4,27	2,07	4371,23	43,71	4,12	2,03
	HASH	MD5	128	4326,27	43,26	7,00	2,65				
		SHA	160	4347,01	43,47	6,50	2,55				
			256	4388,10	43,88	8,62	2,94				
			512	4398,90	43,99	8,42	2,90				

Tabela 4.4: Resultados referente ao consumo de memória (em segundos) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 1 GB para o Raspberry Pi

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s
1 GB	AES	ECB	128	258716,00	2587,16	4116,78	64,16	259068,00	2590,68	3875,57	62,25
			192	257236,00	2572,36	3041,32	55,15	258940,00	2589,40	4797,86	69,27
			256	257900,00	2579,00	4198,59	64,80	258884,00	2588,84	3957,75	62,91
		CBC	128	260052,00	2600,52	3303,32	57,47	259460,00	2594,60	3405,05	58,35
			192	259372,00	2593,72	2621,82	51,20	258936,00	2589,36	2661,24	51,59
			256	258908,00	2589,08	2966,58	54,47	259732,00	2597,32	2890,04	53,76
		CTR	128	258968,00	2589,68	2149,23	46,36	260208,00	2602,08	3105,12	55,72
			192	259572,00	2595,72	3618,02	60,15	260516,00	2605,16	3128,01	55,93
			256	259172,00	2591,72	3061,42	55,33	260244,00	2602,44	2766,51	52,60
		OFB	128	259840,00	2598,40	3041,29	55,15	259940,00	2599,40	3132,89	55,97
			192	259920,00	2599,20	3023,52	54,99	260288,00	2602,88	3017,08	54,93
			256	259836,00	2598,36	2828,31	53,18	260228,00	2602,28	3614,47	60,12
		CFB	128	259224,00	2592,24	2489,15	49,89	259424,00	2594,24	2802,85	52,94
			192	259860,00	2598,60	2862,67	53,50	259532,00	2595,32	3199,05	56,56
			256	259384,00	2593,84	3308,42	57,52	260888,00	2608,88	3709,28	60,90
	DES	ECB	56	258232,00	2582,32	4463,90	66,81	257972,00	2579,72	3710,14	60,91
		CBC	56	256544,00	2565,44	4311,12	65,66	257168,00	2571,68	4165,71	64,54
		OFB	56	257932,00	2579,32	3846,48	62,02	257028,00	2570,28	4293,25	65,52
		CFB	56	256692,00	2566,92	3593,00	59,94	257208,00	2572,08	3846,13	62,02
	3DES	CBC	168	258896,00	2588,96	4781,61	69,15	258416,00	2584,16	3789,23	61,56
		OFB	168	259224,00	2592,24	4820,31	69,43	259200,00	2592,00	4525,25	67,27
		CFB	168	259384,00	2593,84	4405,15	66,37	259140,00	2591,40	3977,82	63,07
	RC4		128	257244,00	2572,44	2437,46	49,37	257968,00	2579,68	3600,70	60,01
	RSA		1024	257780,00	2577,80	8575,15	92,60	254412,00	2544,12	5146,65	71,74
			2048	259368,00	2593,68	8791,33	93,76	256660,00	2566,60	6980,97	83,55
			4096	260776,00	2607,76	7259,42	85,20	256428,00	2564,28	7544,97	86,86
			8192	264052,00	2640,52	7473,67	86,45	255028,00	2550,28	6648,65	81,54
			16384	271072,00	2710,72	6747,44	82,14	259340,00	2593,40	6492,57	80,58
	HASH	MD5	128	250744,00	2507,44	2881,62	53,68				
		SHA	160	251360,00	2513,60	3705,86	60,88				
			256	253012,00	2530,12	3016,55	54,92				
			512	252804,00	2528,04	2639,35	51,37				

4.4 Simulação de Ataques em Cenários Internet das Coisas

Para além dos testes de performance feitos a algoritmos criptográficas, este trabalho procurou também obter uma ideia do comportamento da plataforma utilizada quando regras de *firewall* eram configuradas no seu sistema operativo ou enquanto executava um IDS. Esta experiência recorreu, para isso a simulação de ataques. Para configurar e executar regras de *firewall* foi utilizado `iptables`, enquanto que o *Snort* foi o IDS escolhido. Estas tecnologias foram testadas num *Raspberry Pi 3 (Cortex A53 Quad Core, ARM Cortex, 1.2 GHz, 16 GB de armazenamento, memória de 1 GB com o sistema operativo Ubuntu MATE 16.04.2)*.

Dividiram-se estas simulações em três cenários principais. O primeiro cenário (relatado na sec-

ção 4.4.3) foi onde foram obtidos os resultados para o dispositivo IdC com o *iptables* instalado e configurado. O segundo cenário (secção 4.4.4) foi onde foram obtidos os resultados do dispositivo IdC com o *Snort* instalado e configurado. O último cenário (ver secção 4.4.5) era aquele em que não havia nenhuma destas tecnologias em utilização. O ataque simulado nos referidos cenários foi sempre o `echo-chargen`, utilizando para isso a ferramenta de testes de *stress* para redes `hping3`. As subsecções 4.4.1 e 4.4.2 explicam o ataque e a ferramenta que foi usada, respetivamente.

4.4.1 *echo-chargen*

O `echo-chargen` é um ataque que faz uso dos serviços de rede *echo* e *chargen*. O serviço *echo*, quando ativado numa máquina é um programa que devolve exatamente o mesmo pacote de dados à máquina que a si lhe enviar uma mensagem. O serviço *chargen* é um serviço que gera um pacote aleatório em resposta a uma mensagem que receba. A ideia destes serviços é normalmente ajudar a fazer depuração do estado da rede.

O uso indevido dos recursos de teste do serviço *echo* e *chargen* pode permitir que os atacantes criem cargas de rede mal-intencionadas, falsificando a fonte de transmissão. A forma de operação deste ataque consiste normalmente em (i) identificar uma ou mais máquinas numa rede (ou em redes diferentes) com um ou ambos os serviços em funcionamento; (ii) enviar um pacote com IP e porta fonte falsificados para um dos serviços, declarando que foi o outro serviço (eventualmente noutra máquina) que despoletou o pedido. O primeiro serviço responde para o segundo, que volta a responder para o primeiro, etc. Esta operação coloca as infraestruturas de rede envolvidas num *loop* infinito [Koz03]. O ataque pode causar o estrangulamento da rede ou a paralisação das máquinas envolvidas.

Embora este ataque já não tenha representatividade hoje em dia, é considerado um dos ataques de rede clássicos. Algumas variantes podem voltar a acontecer para outros serviços de teste de rede, e serve bem o propósito dos testes conduzidos no contexto desta dissertação. A figura 4.3 mostra o *output* do software de análise de tráfego de rede *Wireshark* durante ataques efetuados no contexto destas experiências. Como ocorre em ataques `echo-chargen` reais, o endereço IP de origem (vítima) é usado como sendo a fonte e o destino do ataque. Ou seja, parece que a vítima faz um ataque à própria rede e, com isso, sobrecarrega os recursos da mesma.

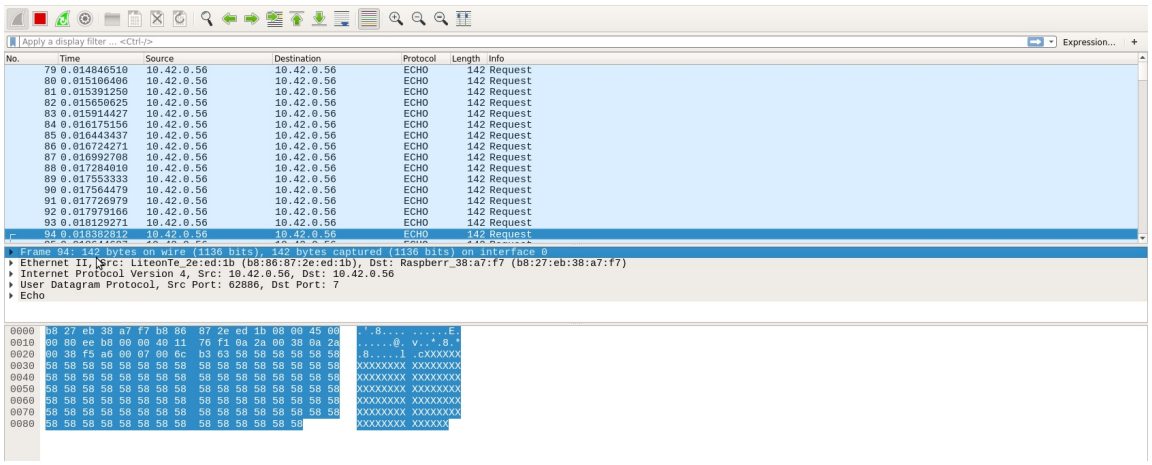

Figura 4.3: Tela do software Wireshark durante o ataque simulado *echo-chargen*.

4.4.2 `hping3`

O `hping3` é uma ferramenta de testes de stresse (usada por atacantes e administradores de rede) para medir a segurança e a resiliência de redes e equipamentos a si ligados. A ferramenta suporta vários protocolos de rede (e.g., *Transmission Control Protocol* (TCP), UDP, *Internet Control Message Protocol* (ICMP)), possui um modo *traceroute*, com a capacidade de enviar pacotes por um canal coberto e muitos outros recursos. Esta ferramenta é útil para as seguintes operações:

- Teste de *firewall*;
- Varrimento avançada de portas;
- Teste de rede, usando diferentes protocolos;
- *Traceroute*, sob todos os protocolos suportados;
- Tempo de atividade remoto;
- Testes de resiliência a *flash crowds* e DDoS;
- Testes de suporte a elevado número de utilizadores e escalabilidade;
- Ferramenta de aprendizagem (usada em contexto laboratorial).

Esta ferramenta foi usada para realização dos ataques para testar o peso da execução das tecnologias de segurança acima descritas (*firewall* e *Snort*) no Raspberry Pi 3 aquando de um ataque. Para isso, foi instrumentalizada como se mostra na figura 4.4 e a seguir:
`hping3 -2 --flood -s 19 -p 7 -d 1000 -a 10.42.0.56 10.42.0.56`.
O comando anterior pode ser decomposto e melhor explicado da seguinte forma:

- `hping3`: ferramenta de testes;
- `-2`: opção para gerar pacotes UDP;
- `--flood`: opção para enviar pacotes o mais rápido possível;
- `-s`: opção para porta de origem;
- `-p`: opção para porta de destino;
- `-d`: tamanho do pacote a ser enviado;
- `-a`: opção para o endereço IP de origem e a seguir o IP de destino.

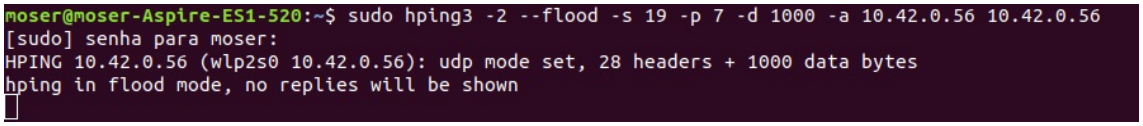

Figura 4.4: Ataque *echo-chargen* usando o *hping3*.

Da análise da figura, pode-se concluir que o ataque é despoletado do computador pessoal ligado à mesma rede local do Raspberry Pi 3.

4.4.3 Testes Usando *Firewall*

Como foi mencionado no capítulo anterior (capítulo 3), a *firewall* é um sistema de defesa que isola o domínio local da rede externa e por consequente, executa a política de controlo de acesso entre estes dois domínios. Para os testes deste cenário foi usada a ferramenta `iptables` para configurar o conjunto de filtros do núcleo Linux de modo a funcionar como *firewall*. O *iptables* é uma ferramenta de segurança oficial usado para configurar o *Netfilter*[2]. O *Netfilter* funciona basicamente através da verificação de cada pacote que atravessa as interfaces de rede e compara-os com um conjunto de regras predefinidas. Estas regras definem as características que estes pacotes devem possuir para que correspondam às ações que serão tomadas nestes pacotes.

Neste cenário (com a instalação e configuração da *firewall*), os testes foram divididos em dois outros sub-cenários. Realizaram-se os testes com a implementação de regras específicas para o ataque em estudo e com regras para outros ataques (não específicas ao `echo-chargen`). A ideia era perceber se a simples inclusão de regras era já suficiente para causar atrasos no processamento do Raspberry Pi 3. Contudo, estes testes demonstraram um consumo de memória e uma utilização da CPU não muito distintos. Com a utilização da CPU de 38% e o consumo de memória de 15.7% equivalendo a 150 MB de consumo como mostra a figura 4.5.

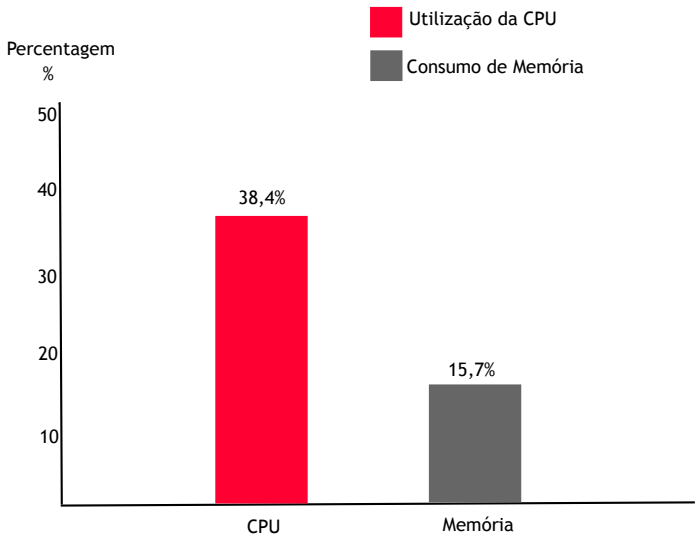

Figura 4.5: Resultado do teste ao Raspberry Pi após configuração da *firewall*.

4.4.4 Testes Usando o *Snort*

O *Snort* é um IDS popular de código aberto. Não é usado apenas para emitir relatórios sobre o tráfego de pacotes de dados anormais dentro da rede, mas também pode ser usado para realizar ações preventivas no sistema. Devido à sua eficiência em detetar ataques direcionados através das regras estabelecidas, pode ser usado tanto nas redes convencionais como em redes mais restritas como é o caso da IdC. As regras do *Snort* são de leitura e configuração simples (por exemplo, a regra usada para detetar ataques `echo-chargen` é `alert udp any 19 <> any 7 (msg:"DOS UDP echo+chargen bomb"; reference:cve,1999-0103; reference:cve,1999-0635; classtype:attempted-dos; sid:271; rev:5;)`).

[2]https://netfilter.org/

No cenário onde o *Snort* se encontrava instalado e configurado, os testes demonstraram um consumo superior (e.g., CPU, memória e Armazenamento) quando comparados com os resultados obtidos a partir dos testes anteriores. A utilização da CPU está na ordem dos 42%, enquanto o consumo de memória se encontra na ordem dos 28%, equivalente a 280 MB de consumo da memória como mostra a figura 4.6. Já em relação ao armazenamento, o *Snort* vários arquivos de *log* que podem ser usados para um estudo minucioso do que acontece na rede e posteriormente tomar as devidas medidas de proteção. Estes *logs*, quando ignorados, podem em fração de minutos ou horas preencher a capacidade de armazenamento disponível nos dispositivos, como é enfatizado na figura 4.7.

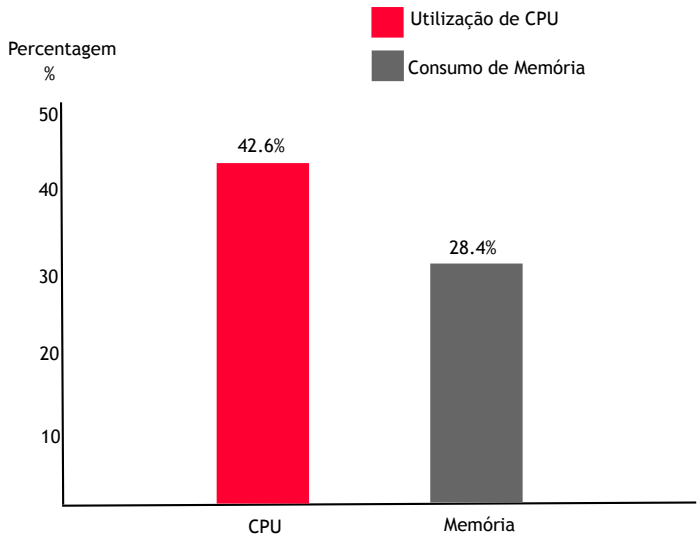

Figura 4.6: Resultado do teste ao Raspberry Pi com o *Snort* instalado.

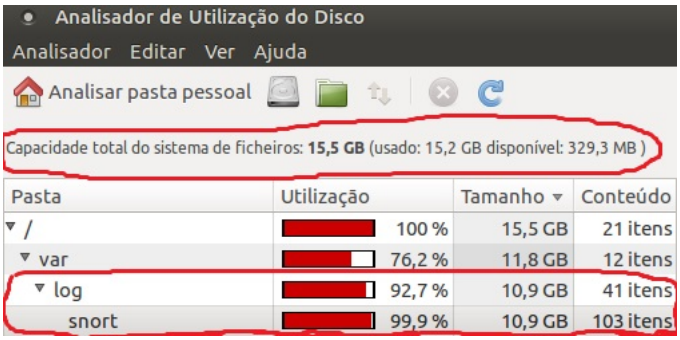

Figura 4.7: Espaço em disco ocupado pelos logs gerados pelo *Snort*.

4.4.5 Cenário sem a *Firewall nem Snort*

Neste último cenário, onde nenhuma das tecnologias de segurança antes referidas foi usada, os testes demonstraram um funcionamento não muito significativo, mas com a utilização considerável da CPU na faixa dos 15% aquando do ataque. Em relação à memória, é apresentado um consumo relativamente inferior quando comparado com os resultados obtidos nos outros cenários na ordem dos 9.5%, como mostra a figura 4.8, o que corresponde ao consumo de 95 MB da memória disponibilizada. Para o Raspberry Pi, estes valores demonstram um consumo não muito comum conhecendo a realidade destes dispositivos, especificamente no que concerne as

limitações de recursos (processamento e memória).

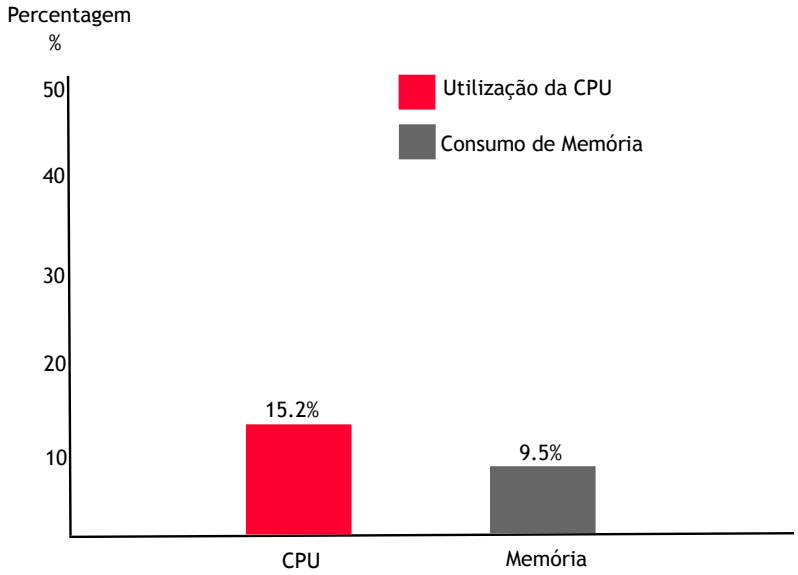

Figura 4.8: Resultado do teste ao Raspberry Pi sem mecanismos de segurança.

4.5 Arquitetura para Prova de Conceito

A presente secção descreve uma arquitetura idealizada para uma ferramenta de segurança que envolva o mapeamento apresentado neste documento e que de modo especial poderá ser útil em reconhecer as necessidades de segurança apresentadas pelos dispositivos IdC. O objetivo desta arquitetura ou da ferramenta que a vai implementar é facilitar a escolha de mecanismos específicos de segurança, com base nas limitações que os dispositivos apresentam. A subsecção 4.5.1 apresenta uma breve descrição da arquitetura da ferramenta proposta para prova de conceito dos requisitos e os mecanismos de segurança apresentados neste documento.

4.5.1 Descrição da Arquitetura

Apesar das incertezas associadas ao campo de estudo em análise, há vários factos que podem imediatamente ser tomados em consideração: (i) não é sensato pensar que os programadores ou arquitetos de sistemas das IdC venham a saber mais de segurança num futuro próximo; (ii) o conjunto de mecanismos e tecnologias disponíveis para a IdC é algo muito dinâmico; (iii) é óbvio que é necessário começar a pensar a segurança desde a conceção do produto. Assim, é necessário construir ferramentas que ajudem nesse processo (o da conceção), mas que sejam capazes de absorver novas tecnologias à medida que aparecem.

A arquitetura brevemente descrita nesta secção foi elaborada num trabalho paralelo no mesmo laboratório e contexto de investigação deste trabalho de investigação e também foi apresentada em [SSFI18]. Foi decidido mencioná-la aqui, pois a ferramenta que pode vir a representar irá muito provavelmente usar parte dos resultados aqui apresentados. A arquitetura está esquematizada na figura 4.9 e é composta fundamentalmente por três partes nomeadamente: *User Query, Security Manager, Database*.

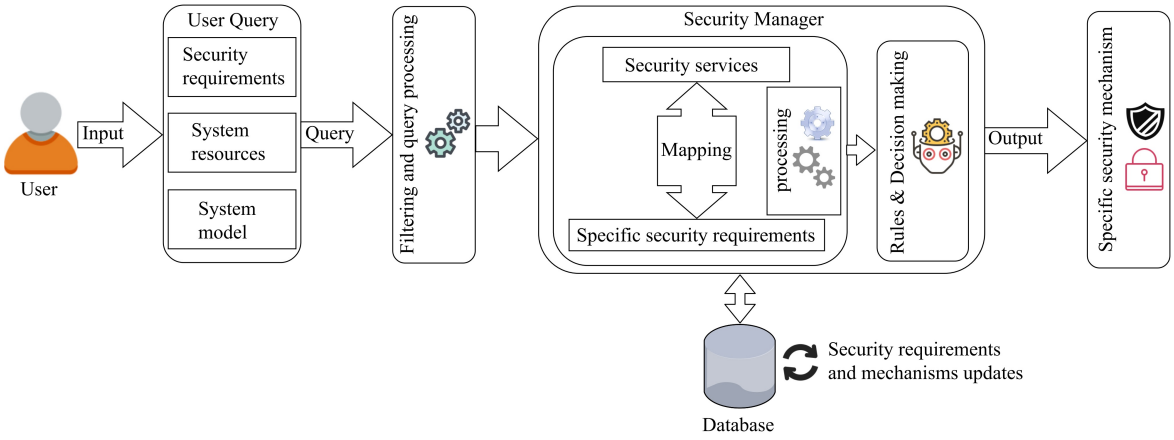

Figura 4.9: Arquitetura básica da ferramenta de mapeamento de requisitos e mecanismos/tecnologias de segurança, ainda geral e na forma de prova de conceito (adaptada de [SSFI18]).

A parte *User Query* representa a componente que irá permitir estruturar a entrada fornecida pelo utilizador e composta por *Security requeriments, System resources e System model*. Estas várias informações devem ser eventualmente o *output* de outra ferramenta de identificação e formação de requisitos. Na *query*, o utilizador introduz um conjunto de requisitos de segurança e características específicas dos dispositivos IdC, que são filtrados e enviados para a componente *Security Manager* que, por sua vez, tem a missão de selecionar e mapear os requisitos filtrados através de uma consulta efetuada a uma base de dados onde são armazenados todos os requisitos e mecanismos, atualizados periodicamente desde que haja necessidade para tal. Estes requisitos são escolhidos de acordo o *System model* e ao aplicar um conjunto de regras de acordo os padrões dos recursos dos dispositivos. A saída deve ser uma lista de mecanismos de segurança específicos para o dispositivo.

4.6 Conclusões

No presente capítulo foram apresentados um conjunto de testes efetuados no decurso do projeto de dissertação a alguns dos mecanismos de segurança disponíveis hoje em dia para sistemas computacionais, nomeadamente algoritmos de cifra, IDSs e tecnologia *firewall*. Estes testes foram feitos numa plataforma tipicamente associada à IdC com o intuito de perceber melhor o impacto de importar diretamente as tecnologias para este novo ambiente. Incluiu-se, ainda, uma breve descrição da arquitetura básica da ferramenta proposta para o mapeamento dos requisitos e mecanismos de segurança específicos para a IdC. Esta ferramenta poderá vir a ser usada por utilizadores com pouca ou experiência na área da segurança aquando do desenho de novas soluções para a IdC, ou que que estejam a procurar melhorar os seus conhecimentos neste ramo.

Os resultados obtidos dos testes aos algoritmos criptográficos no Raspberry Pi 3, quando comparados com os resultados obtidos num computador pessoal, demonstraram maior consumo de tempo e um menor consumo de memória, o que augura a favor da viabilidade da utilização da implementação de alguns destes algoritmos em plataformas de recursos mais limitados. Contudo, a percentagem de recursos utilizada também demonstrou que a combinação de vários destes mecanismos de segurança pode não ser possível nestes ou em dispositivos ainda mais li-

mitados. Nos resultados destacaram-se o RC4, AES e o RSA em termos de performance, e os dois últimos como globalmente os mais indicados para utilização em cenários da IdC com recursos semelhantes aos representados pelo Raspberry Pi 3. Algumas implementações (e.g., do 3DES) não executavam com sucesso para ficheiros com 2 GB, o que constitui um resultado curioso. No entanto, podemos notar um decréscimo de performance quando alteramos os ficheiros de 100 MB para ficheiros de 1 GB e 2 GB, respetivamente, facto que pode ser usado em contexto de ataque para efeitos de DoS/DDoS.

Em relação aos testes realizados para as funcionalidades de *firewall* e IDS, os resultados demonstraram que estes dois mecanismos de segurança necessitam de uma abordagem dirigida única e exclusivamente para os dispositivos da IdC. Os resultados de desempenho demonstram que as tecnologias podem ser colocadas em funcionamento num Rasperry Pi 3, mas com impacto significativo no poder de processamento e memória durante um ataque. A implementação da regra na *firewall* não causou, por outro lado, grandes variações ao desempenho. No caso do *Snort*, ficou claro que deve ser dada atenção ao local onde os logs são guardados, pois rapidamente podem esgotar o espaço da memória persistente destes dispositivos.

Capítulo 5

Conclusão e Trabalho Futuro

5.1 Principais Conclusões

A Internet está próxima de ser considerada um bem indispensável em sociedades desenvolvidas e a IdC adicionou um novo potencial à Internet, permitindo a comunicação entre objetos e humanos de uma forma imersiva, criando uma sociedade cada vez mais inteligente e mais dependente destes meios tecnológicos. Isto permitiu ainda a materialização da visão das comunicações realizadas a qualquer hora e qualquer lugar e por diferentes meios. A IdC ir-se-á rapidamente tornar a parte central da Internet, em termos de utilidade, tendo em conta o rumo que tem tomado, envolvendo a sociedade como peça fundamental e indispensável para o seu crescimento e como tecnologia voltada para a resolução de problemas que afligem a sociedade de modo geral.

O estudo levado a cabo durante o trabalho de dissertação deixou claro que novas medidas e mecanismos de segurança são necessários para manter a IdC no caminho certo e para que esta cumpra os seus objetivos. Entretanto, de acordo as investigações realizadas e apresentadas nesta dissertação, percebe-se que os dispositivos da IdC, nomeadamente as coisas, são bastantes vulneráveis, com fraca integração de mecanismos ou boas práticas de segurança. Dada a sua franca adoção, ainda a acelerar, é de extrema importância o desenvolvimento de técnicas ou tecnologias que estejam direcionadas especificamente para a segurança destes dispositivos, não só devido ao ambiente que envolve aplicações da IdC, totalmente diferente do que existe atualmente, como também devido ao facto de que não chega transpor para a IdC todos aqueles mecanismos existentes para redes ou sistemas atuais.

Esta dissertação, primeiramente, apresentou vários conceitos e definições envolvendo o mundo da IdC, procurando abordar as questões pertinentes de forma simples e ilustrativa para o melhor entendimento das mesmas. Além de abordar as questões de segurança voltadas para os dispositivos inteligentes, apresentou um mapeamento entre requisitos e mecanismos de segurança para a IdC, e discutiu um conjunto de testes de performance a algoritmos da criptografia moderna muito utilizadas em soluções de segurança informática, bem como de duas tecnologias de *firewall* e IDS. Estes testes foram efetuados num dispositivo específico para a IdC, o Raspberry Pi 3. Os resultados desta performance são comparados com os resultados obtidos num computador pessoal. Estas contribuições constituem apenas uma parte do longo caminho a percorrer para uma IdC mais segura por desenho. Acredita-se que o estudo e relato do comportamento dos algoritmos e mecanismos de segurança em dispositivos da IdC ajuda a um melhor entendimento dos problemas e limitações, bem como na forma de pensar a segurança para a IdC no futuro e auxilia a tomada de decisões aquando da escolha de mecanismos a implementar.

Enquanto que o desenvolvimento e integração de mecanismos de segurança ainda é feito sobretudo de forma reativa, i.e., após os desastres ou problemas aparecerem, é de esperar que

a longo prazo a segurança seja garantida por construção. No caso da IdC, e conforme prova esta dissertação, ainda existe muito trabalho a ser feito para que isso aconteça. O cenário vai melhorar à medida que a tecnologia melhora (mais poder de processamento nos equipamentos *coisas*), à medida que a maturidade e estagnação das aplicações da IdC aumenta e à medida que os programadores, arquitetos de sistemas e utilizadores se tornam mais conscientes. Até lá, será necessário abordar os problemas com caráter de urgência e de recurso. Apesar ser verdade que muitos dispositivos da IdC não têm capacidade para integrarem mecanismos de segurança (ou mesmo para combinações de mais do que um destes mecanismos), também é verdade que o estado da segurança podia estar muito melhor se mecanismos existentes fossem de facto integrados em soluções IdC.

5.2 Trabalho Futuro

Como trabalho futuro, ir-se-á dar continuidade a este estudo, nomeadamente através do alargamento dos testes de performance a um conjunto ainda maior de mecanismos e tecnologias de segurança existentes, assim como a diferentes plataformas da IdC como Arduino ou sensores. Durante os testes futuros, será também desejável a medição do consumo energético dos dispositivos durante as operações, tendo em consideração que estes dispositivos atuam, muitas vezes, com fornecimento de energia limitado. Poder-se-á também enveredar pela análise e descrição de ataques recentes e específicos à IdC, na tentativa de isolar comportamentos específicos que possam ajudar na sua prevenção.

Durante a dissertação foram identificados vários pontos de investigação futura a ser endereçados, para além daqueles em que a investigação científica se tem focado (e.g., mecanismos leves de criptografia). Um desses pontos, por exemplo, refere-se à necessidade de desenvolver políticas de segurança para cenários da IdC. Poder-se-á pensar abordar esta questão por via do desenho e desenvolvimento de uma ferramenta que guie administradores de sistemas, programadores, gestores ou utilizadores na elaboração desses documentos a partir de modelos.

Será também necessário o desenvolvimento da ferramenta de mapeamento para a qual a arquitetura foi descrita nesta dissertação. Essa ferramenta, direcionada para programadores ou arquitetos de sistema, deverá ser escalável e de simples interação.

Bibliografia

[ADAS17] Waqar Ali, Ghulam Dustgeer, Muhammad Awais, and Munam Ali Shah. IoT based Smart Home : Security Challenges , Security Requirements and Solutions. In 2017 23rd International Conference on Automation and Computing (ICAC), pages 1–6, Huddersfield, UK, 2017. IEEE. Available from: http://ieeexplore.ieee.org/document/8082057/. 20, 31

[Alq17] Fayez Hussain Alqahtani. Developing an Information Security Policy: A Case Study Approach. Procedia Computer Science, 124:691–697, 2017. Available from: https://doi.org/10.1016/j.procs.2017.12.206. 40

[AOHA17] Fadele Ayotunde Alaba, Mazliza Othman, Ibrahim Abaker Targio Hashem, and Faiz Alotaibi. Internet of things security: A survey. Journal of Network and Computer Applications, 88:10 – 28, 2017. Available from: http://www.sciencedirect.com/science/article/pii/S1084804517301455. 12, 33

[AP15] Alex Akinbi and Ella Pereira. Mapping security requirements to identify critical security areas of focus in PaaS cloud models. Proceedings - 15th IEEE International Conference on Computer and Information Technology, CIT 2015, pages 789–794, 2015. Available from: http://ieeexplore.ieee.org/document/7363156/. 20, 25

[ATANM+16] Majid A. Al-Taee, Waleed Al-Nuaimy, Zahra J. Muhsin, Ali Al-Ataby, and Ahmad M. Al-Taee. Mapping Security Requirements of Mobile Health Systems into Software Development Lifecycle. 2016 9th International Conference on Developments in eSystems Engineering (DeSE), pages 87–93, 2016. Available from: http://ieeexplore.ieee.org/document/7930629/. 20, 25

[ATANMAA16] Majid A. Al-Taee, Waleed Al-Nuaimy, Zahra J. Muhsin, and Ali Al-Ataby. Robot Assistant in Management of Diabetes in Children Based on the Internet of Things. IEEE Internet of Things Journal, 4(2):437–445, 2016. Available from: http://ieeexplore.ieee.org/document/7728003/. 20

[ATN17] M. Anirudh, S. A. Thileeban, and D. J. Nallathambi. Use of honeypots for mitigating dos attacks targeted on iot networks. In 2017 International Conference on Computer, Communication and Signal Processing (ICCCSP), pages 1–4, Jan 2017. 38

[CM10] Floerkemeier Christian and Friedemann Mattern. From Active Data Management to Event-Based Systems and More, chapter From the Internet of Computers to the Internet of Things, pages 242–259. Springer Berlin Heidelberg 2010, 2010. 10

[CMA+13] Tein-Yaw Chung, Ibrahim Mashal, Osama Alsaryrah, Van Huy, Wen-Hsing Kuo, and Dharma P. Agrawal. Social Web of Things: A Survey. In 2013 International Conference on Parallel and Distributed Systems, pages 570–575, Seoul, South Korea, 2013. IEEE. Available from: http://ieeexplore.ieee.org/document/6808239/. 8

[CWG+14] M. Chen, J. Wan, S. Gonzalez, X. Liao, and V. C. M. Leung. A survey of recent developments in home m2m networks. IEEE Communications Surveys Tutorials, 16(1):98–114, First 2014. 9

[dKH17] S. d. Krit and E. Haimoud. Overview of firewalls: Types and policies: Managing windows embedded firewall programmatically. In 2017 International Conference on Engineering MIS (ICEMIS), pages 1–7, May 2017. 34

[DKS17] Maryam Daud, Quratulain Khan, and Yasir Saleem. A Study of Key Technologies for IoT and associated Security Challenges. 2017 IEEE International Symposium on Wireless Systems and Networks (ISWSN), 2017. Available from: http://ieeexplore.ieee.org/document/8250042/. 21

[HCLY16] Xin Huang, Paul Craig, Hangyu Lin, and Zheng Yan. Seciot: a security framework for the internet of things. Security and Communication Networks, 9(16):3083–3094, 2016. SCN-14-0857.R1. Available from: http://dx.doi.org/10.1002/sec.1259. 22, 25

[HFH15] M. M. Hossain, M. Fotouhi, and R. Hasan. Towards an analysis of security issues, challenges, and open problems in the internet of things. In 2015 IEEE World Congress on Services, pages 21–28, June 2015. Available from: https://doi.org/10.1109/SERVICES.2015.12. 22, 25

[Hua16] Kien A. Hua. Internet of Things: Challenges and opportunities for collaborative technologies (invited Talk). Proceedings - 2016 International Conference on Collaboration Technologies and Systems, CTS 2016, pages 613–614, 2016. Available from: http://ieeexplore.ieee.org/document/7871052/. 12

[IW17] Dimas Dwiki Ismoyo and Rini Wisnu Wardhani. Block cipher and stream cipher algorithm performance comparison in a personal VPN gateway. Proceedings - 2016 International Seminar on Application of Technology for Information and Communication, ISEMANTIC 2016, pages 207–210, 2017. 23, 25

[Joh14] Stig Tore Johannesen. Cryptoprocessing on the Arduino. Technical Report August, Norwegian University of Science and Technology, 2014. Available from: https://brage.bibsys.no/xmlui/bitstream/handle/11250/2352224/9554{_}FULLTEXT.pdf?sequence=1. 24, 25

[JVW+14] Qi Jing, Athanasios V. Vasilakos, Jiafu Wan, Jingwei Lu, and Dechao Qiu. Security of the internet of things: perspectives and challenges. Wireless Networks, 20(8):2481–2501, Nov 2014. Available from: https://doi.org/10.1007/s11276-014-0761-7. 21, 25

[KH17] N. Kaliya and M. Hussain. Framework for privacy preservation in iot through classification and access control mechanisms. In 2017 2nd International Conference for Convergence in Technology (I2CT), pages 430–434, April 2017. Available from: https://doi.org/10.1109/I2CT.2017.8226166. 23, 25

[Kim15] Jung Tae Kim. Requirement of security for IoT application based on gateway. International Journal of Security and its Applications, 9(10):201–208, 2015. Available from: http://dx.doi.org/10.14257/ijsia.2015.9.10.18. 21, 25

[KKZK12] Rafiullah Khan, Sarmad Ullah Khan, Rifaqat Zaheer, and Shahid Khan. Future internet: The internet of things architecture, possible applications and key challenges. Proceedings - 10th International Conference on Frontiers of Information Technology, FIT 2012, 34:257–260, 2012. 14

[KM17] Tinku Kumar and P. B. Mane. ZigBee topology: A survey. 2016 International Conference on Control Instrumentation Communication and Computational Technologies, ICCICCT 2016, pages 164–166, 2017. Available from: http://ieeexplore.ieee.org/document/7987937/. 16

[KMAM16] Joona Kannisto, Niko Makitalo, Timo Aaltonen, and Tommi Mikkonen. Programming Model Perspective on Security and Privacy of Social Cyber-physical Systems. 2016 IEEE International Conference on Mobile Services (MS), pages 87–94, 2016. Available from: http://ieeexplore.ieee.org/document/7787059/. 33

[Koz03] J. Koziol. Intrusion Detection with Snort. Sams, 2003. Available from: https://books.google.pt/books?id=sVqEFXjbcRoC. 55

[KPR16] M. Kuzlu, M. Pipattanasomporn, and S. Rahman. Review of communication technologies for smart homes/building applications. Proceedings of the 2015 IEEE Innovative Smart Grid Technologies - Asia, ISGT ASIA 2015, pages 4–9, 2016. 18

[KT17] A. Kaushik and S. Talati. Securing iot using layer characterstics. In 2017 3rd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), pages 290–298, Dec 2017. 11

[Lar17] Selena Larson. Fda confirms that st. jude's cardiac devices can be hacked [online]. 2017. Available from: http://money.cnn.com/2017/01/09/technology/fda-st-jude-cardiac-hack/ [cited 18 Junho 2018]. 26

[LF07] Jun-Dian Lee and Chih-Peng Fan. Efficient low-latency rc4 architecture designs for ieee 802.11i wep/tkip. In 2007 International Symposium on Intelligent Signal Processing and Communication Systems, pages 56–59, Nov 2007. 46

[LG17] Rui Liu and Yongqi Ge. Smart Home System Design Based on Internet of Things. In The 12th International Conference on Computer Science & Education (ICCSE 2017) August 22-25, 2017. University of Houston, USA, pages 444–448, Houston, TX, USA, 2017. IEEE. Available from: http://ieeexplore.ieee.org/document/8085533/. 14

[LL16] Phillip A Laplante and Nancy Laplante. The Internet of Things in Healthcare: Potential Applications and Challenges. IT Professional Magazine, 18(3):2, 2016. 14

[LXC12] Jing Liu, Yang Xiao, and C.L. Philip Chen. Authentication and Access Control in the Internet of Things. In 2012 32nd International Conference on Distributed Computing Systems Workshops, pages 588–592, Macau, China, 2012. IEEE. Available from: http://ieeexplore.ieee.org/document/6258209/. 32

[LXW11] Xiong Li, Zhou Xuan, and Liu Wen. Research on the architecture of trusted security system based on the internet of things. Proceedings - 4th International Conference on Intelligent Computation Technology and Automation, ICICTA

2011, 2:1172–1175, 2011. Available from: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp={&}arnumber=5751104. 38

[LYZ+17] Jie Lin, Wei Yu, Nan Zhang, Xinyu Yang, Hanlin Zhang, and Wei Zhao. A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications. IEEE Internet of Things Journal, 4(5):1–1, 2017. Available from: http://ieeexplore.ieee.org/document/7879243/. 13, 21, 25, 29

[MAH16] Konstantinos Markantonakis, Raja Naeem Akram, and Royal Holloway. A secure and trusted boot process for Avionics Wireless Networks. In ICNS 2016: Securing an Integrated CNS System to Meet Future Challenges, pages 1–9, Herndon, VA, USA, 2016. IEEE. 34

[MD16] Somayya Madakam and Hema Date. Security mechanisms for connectivityof smart devices in the internet of things. In Zaigham Mahmood, editor, Connectivity Frameworks for Smart Devices, chapter 2, pages 23–41. Springer International Publishing Switzerland 2016, Derby, UK, 2016 edition, 2016. Available from: http://link.springer.com/10.1007/978-3-319-33124-9. 40

[Med16] Aref Meddeb. Internet of things standards: who stands out from the crowd? IEEE, 54(7):40–47, 2016. Available from: http://ieeexplore.ieee.org/document/7514162/. 1, 7

[MHFH16] Lukas Malina, Jan Hajny, Radek Fujdiak, and Jiri Hosek. On perspective of security and privacy-preserving solutions in the internet of things. Computer Networks, 102(Supplement C):83 – 95, 2016. Available from: http://www.sciencedirect.com/science/article/pii/S1389128616300779. 23, 25

[MO13] Catherine E A Mulligan and Magnus Olsson. Architectural implications of smart city business models: An evolutionary perspective. IEEE Communications Magazine, 51(6):80–85, 2013. Available from: http://ieeexplore.ieee.org/document/6525599/. 11

[MPT12] A. K. Mandal, C. Parakash, and A. Tiwari. Performance evaluation of cryptographic algorithms: Des and aes. In Electrical, Electronics and Computer Science (SCEECS), 2012 IEEE Students' Conference on, pages 1–5, March 2012. 46

[MPY17] Diego M. Mendez, Ioannis Papapanagiotou, and Baijian Yang. Internet of things: Survey on security and privacy. CoRR, abs/1707.01879, 2017. Available from: http://arxiv.org/abs/1707.01879. 22, 25

[MRON17] Gift Matsemela, Suvendi Rimer, Khmaies Ouahada, and Richard Ndjiongue. Internet of Things Data Integrity. In IEEE, editor, IST-Africa Week Conference (IST-Africa), 2017, pages 1–9, Windhoek, Namibia, 2017. IEEE. Available from: http://ieeexplore.ieee.org/document/8102332/. 23, 25

[Nal16] Shayan Nalbandian. A survey on Internet of Things: Applications and challenges. 2nd International Congress on Technology, Communication and Knowledge, ICTCK 2015, pages 165–169, 2016. Available from: http://ieeexplore.ieee.org/document/7582664/. 10

[Nor16] A. Nordrum. The internet of fewer things [news]. IEEE Spectrum, 53(10):12–13, October 2016. Available from: https://doi.org/10.1109/MSPEC.2016.7572524. xiii, 10, 43

[NOSALS18] J. Navarro-Ortiz, S. Sendra, P. Ameigeiras, and J. M. Lopez-Soler. Integration of lorawan and 4g/5g for the industrial internet of things. IEEE Communications Magazine, 56(2):60–67, Feb 2018. 18

[OK17] S. R. Oh and Y. G. Kim. Security requirements analysis for the iot. In 2017 International Conference on Platform Technology and Service (PlatCon), pages 1–6, Feb 2017. Available from: http://ieeexplore.ieee.org/document/7883727/. 21, 25

[OMEO17] Aafaf Ouaddah, Hajar Mousannif, Anas Abou Elkalam, and Abdellah Ait Ouahman. Access control in the internet of things: Big challenges and new opportunities. Computer Networks, 112:237 – 262, 2017. Available from: http://www.sciencedirect.com/science/article/pii/S1389128616303735. 36

[PHV17] S. Pal, M. Hitchens, and V. Varadharajan. On the design of security mechanisms for the internet of things. In 2017 Eleventh International Conference on Sensing Technology (ICST), pages 1–6, Dec 2017. Available from: https://doi.org/10.1109/ICSensT.2017.8304476. 22

[PK17] Jaemin Park and Kwangjo Kim. TM-Coin: Trustworthy management of TCB measurements in IoT. 2017 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2017, pages 654–659, 2017. Available from: http://ieeexplore.ieee.org/document/7917640/. 39

[REU16] REUTERS. J&j warns diabetic patients: Insulin pump vulnerable to hacking [online]. 2016. Available from: https://www.reuters.com/article/us-johnson-johnson-cyber-insulin-pumps-e/jj-warns-diabetic-patients-insulin-pump-vulnerable-to-hacking-idUSKCN12411L [cited 18 Junho 2018]. 26

[RPK11] A. J Dinusha Rathnayaka, Vidyasagar M. Potdar, and Samitha J. Kuruppu. Evaluation of wireless home automation technologies. 5th IEEE International Conference on Digital Ecosystems and Technologies (IEEE DEST 2011), 5:76–81, 2011. Available from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5936601. 19

[RSA78] R.L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21:120–126, 1978. 47

[RSS+12] B B P Rao, P Saluia, N Sharma, A Mittal, and S V Sharma. Cloud computing for Internet of Things & sensing based applications. In Sensing Technology (ICST), 2012 Sixth International Conference on, pages 374–380, 2012. Available from: http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6461705{%}5Cnpapers3://publication/doi/10.1109/ICSensT.2012.6461705. 12

[SA11] P. Saint-Andre. Extensible messaging and presence protocol (xmpp): Core [online]. 2011. Available from: https://xmpp.org/rfcs/rfc6120.html#RFC3921 [cited 17 Abril 2018]. 18

[SC15] M. Schukat and P. Cortijo. Public key infrastructures and digital certificates for the internet of things. In 2015 26th Irish Signals and Systems Conference (ISSC), pages 1–5, June 2015. 36

[SKB+08] Kevin Stine, Rich Kissel, William C Barker, Annabelle Lee, and Jim Fashlsing. SP 800-60 Volume II : Appendices to Guide for Mapping Types of Information and Information Systems to Security Categories. National Institute of Standards and Technology, II(August), 2008. 29

[SKL17] Mangal Sain, Young Jin Kang, and Hoon Jae Lee. Survey on security in Internet of Things: State of the art and challenges. 2017 19th International Conference on Advanced Communication Technology (ICACT), pages 699–704, 2017. Available from: http://ieeexplore.ieee.org/document/7890183/. 15

[SRGCP15] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini. Security, privacy and trust in Internet of things: The road ahead. Computer Networks, 76:146–164, 2015. Available from: http://dx.doi.org/10.1016/j.comnet.2014.11.008. 31

[SSC+18] Musa G Samaila, João B. F Sequeiros, Acácio F. P. P Correia, Mário M. Freire, and Pedro R.M Inácio. Iot hardware development platforms: Past, present, and future. In Atta ur Rehman Khan Qusay F. Hassan and Sajjad A. Madani, editors, Internet of Things - Challenges, Advances, and Applications, chapter 6, pages 107–139. Chapman and Hall/CRC, 2018. 45

[SSFI18] Musa G. Samaila, João B. F. Sequeiros, Mário M. Freire, and Pedro R. M. Inácio. Security threats and possible countermeasures in iot applications covering different industry domains. In Proceedings of the 13th International Conference on Availability, Reliability and Security, ARES 2018, pages 16:1–16:9, New York, NY, USA, 2018. ACM. Available from: http://doi.acm.org/10.1145/3230833.3232800. xiv, 59, 60

[SV17] Janit Modi Shivangi Vashi, Jyotsnamayee Ram. Internet of things (iot) a vision, architectural elements, and security issues. IEEE, 1(1):492–496, 2017. xiii, 8

[TMV+14] D. Thangavel, X. Ma, A. Valera, H. X. Tan, and C. K. Y. Tan. Performance evaluation of mqtt and coap via a common middleware. In 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pages 1–6, April 2014. 16

[VDL+16] Emmanouil Vasilomanolakis, Jörg Daubert, Manisha Luthra, Vangelis Gazis, Alex Wiesmaier, and Panayotis Kikiras. On the Security and Privacy of Internet of Things Architectures and Systems. Proceedings - 2015 International Workshop on Secure Internet of Things, SIoT 2015, pages 49–57, 2016. Available from: https://doi.org/10.1109/SIOT.2015.9. 22, 25

[WLL+10] Miao Wu, Ting Jie Lu, Fei Yang Ling, Jing Sun, and Hui Ying Du. Research on the architecture of Internet of Things. ICACTE 2010 - 2010 3rd International Conference on Advanced Computer Theory and Engineering, Proceedings, 5:484–487, 2010. Available from: http://ieeexplore.ieee.org/document/5579493/. 12, 13, 14

[YQ17] F. Ye and Y. Qian. A security architecture for networked internet of things devices. In GLOBECOM 2017 - 2017 IEEE Global Communications Conference, pages 1–6, Dec 2017. 21, 25

[YSAAH17] M. B. Yassein, M. Q. Shatnawi, S. Aljwarneh, and R. Al-Hatmi. Internet of things: Survey and open issues of mqtt protocol. In 2017 International Conference on Engineering MIS (ICEMIS), pages 1–6, May 2017. 16

Apêndice A

Resultados dos Testes ao Openssl

Este anexo contém a totalidade dos resultados obtidos durante os testes descritos no capítulo 4, colocados aqui por uma questão de organização desta dissertação. A discussão desses resultados também é feita no mesmo capítulo. O anexo contém as tabelas A.1 a A.12, com resultados de performance computacional dos algoritmos criptográficos analisados no contexto da dissertação para ficheiros de diferentes tamanhos.

Tabela A.1: Resultados referente ao tempo gasto (em segundos) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 100 MB para o Computador Portátil.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
100 MB	AES	ECB	128	92,25	0,92	0,88	0,94	100,06	1,00	1,03	1,01
			192	96,20	0,96	0,97	0,98	97,25	0,97	0,98	0,99
			256	92,15	0,92	0,87	0,93	99,92	1,00	1,03	1,02
		CBC	128	114,12	1,14	1,34	1,16	87,21	0,87	0,78	0,89
			192	113,83	1,14	1,32	1,15	87,02	0,87	0,78	0,88
			256	118,05	1,18	1,42	1,19	81,52	0,82	0,68	0,82
		CTR	128	96,43	0,96	0,95	0,98	98,32	0,98	0,99	0,99
			192	96,28	0,96	0,95	0,98	100,12	1,00	1,03	1,01
			256	97,94	0,98	0,99	1,00	100,24	1,00	1,04	1,02
		OFB	128	96,22	0,96	0,94	0,97	96,36	0,96	0,94	0,97
			192	94,62	0,95	0,90	0,95	95,74	0,96	0,92	0,96
			256	96,66	0,97	0,94	0,97	98,58	0,99	0,98	0,99
		CFB	128	100,73	1,01	1,02	1,01	100,09	1,00	1,01	1,00
			192	105,74	1,06	1,12	1,06	106,89	1,07	1,15	1,07
			256	109,88	1,10	1,21	1,10	111,63	1,12	1,26	1,12
	DES	ECB	56	463,78	4,64	21,52	4,64	458,57	4,59	21,04	4,59
		CBC	56	455,53	4,56	20,76	4,56	447,75	4,48	20,07	4,48
		OFB	56	492,65	4,93	24,28	4,93	496,62	4,97	24,70	4,97
		CFB	56	508,97	5,09	25,92	5,09	511,90	5,12	26,22	5,12
	3DES	CBC	168	1149,50	11,50	132,15	11,50	1155,57	11,56	133,58	11,56
		OFB	168	1192,50	11,93	142,24	11,93	1187,53	11,88	141,04	11,88
		CFB	168	1215,60	12,16	147,79	12,16	1212,55	12,13	147,05	12,13
	RC4		128	97,95	0,98	0,97	0,98	96,23	0,96	0,94	0,97
	RSA		1024	127,81	1,28	3,17E-05	5,63E-03	128,04	1,28	1,80E-05	4,25E-03
			2048	128,37	1,28	2,35E-05	4,85E-03	127,81	1,28	3,17E-05	5,63E-03
			4096	131,91	1,32	3,05E-05	5,52E-03	127,91	1,28	4,46E-05	6,68E-03
			8192	155,35	1,55	3,31E-05	5,75E-03	128,06	1,28	2,19E-05	4,68E-03
			16384	333,28	3,33	6,74E-03	8,21E-02	128,73	1,29	4,21E-05	6,49E-03
	HASH	MD5	128	40,33	0,40	3,45E-05	5,87E-03				
		SHA	160	61,77	0,62	6,84E-05	8,27E-03				
			256	127,90	1,28	3,33E-05	5,77E-03				
			512	103,92	1,04	2,16E-05	4,64E-03				

Tabela A.2: Resultados referente ao consumo de memória (em *Kilobytes*) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 100 MB para o Computador Portátil.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s
100 MB	AES	ECB	128	447684,00	4476,84	4562,33	67,55	447232,00	4472,32	4764,38	69,02
			192	446420,00	4464,20	5071,16	71,21	446276,00	4462,76	4892,38	69,95
			256	446684,00	4466,84	3909,85	62,53	445908,00	4459,08	3675,15	60,62
		CBC	128	445556,00	4455,56	4306,53	65,62	445728,00	4457,28	4785,56	69,18
			192	446156,00	4461,56	5124,77	71,59	446492,00	4464,92	4094,03	63,98
			256	447248,00	4472,48	3957,21	62,91	446596,00	4465,96	4286,56	65,47
		CTR	128	447304,00	4473,04	4501,64	67,09	447288,00	4472,88	4700,35	68,56
			192	446596,00	4465,96	4814,56	69,39	447232,00	4472,32	4497,82	67,07
			256	446452,00	4464,52	4262,61	65,29	447372,00	4473,72	4584,08	67,71
		OFB	128	445800,00	4458,00	3926,88	62,66	446996,00	4469,96	4010,08	63,33
			192	446632,00	4466,32	3035,58	55,10	446492,00	4464,92	4997,71	70,69
			256	445840,00	4458,40	4140,80	64,35	446848,00	4468,48	3897,37	62,43
		CFB	128	445780,00	4457,80	4907,32	70,05	445700,00	4457,00	4832,12	69,51
			192	446856,00	4468,56	4002,25	63,26	447584,00	4475,84	3871,33	62,22
			256	447112,00	4471,12	3736,83	61,13	446768,00	4467,68	3875,10	62,25
	DES	ECB	56	445236,00	4452,36	3501,79	59,18	445528,00	4455,28	5259,32	72,52
		CBC	56	446264,00	4462,64	6117,83	78,22	443792,00	4437,92	4764,95	69,03
		OFB	56	445152,00	4451,52	5358,17	73,20	445300,00	4453,00	4612,60	67,92
		CFB	56	445916,00	4459,16	5602,33	74,85	445056,00	4450,56	5265,45	72,56
	3DES	CBC	168	448928,00	4489,28	5878,68	76,67	448160,00	4481,60	5375,36	73,32
		OFB	168	445324,00	4453,24	4807,58	69,34	446724,00	4467,24	5633,18	75,05
		CFB	168	448264,00	4482,64	5163,59	71,86	449268,00	4492,68	4858,26	69,70
	RC4		128	447336,00	4473,36	5606,47	74,88	447000,00	4470,00	7093,28	84,22
	RSA		1024	463304,00	4633,04	4423,63	66,51	461448,00	4614,48	4742,72	68,87
			2048	462256,00	4622,56	5590,15	74,77	465080,00	4650,80	2387,23	48,86
			4096	461740,00	4617,40	4477,21	66,91	465196,00	4651,96	2377,21	48,76
			8192	464724,00	4647,24	4376,14	66,15	466080,00	4660,80	2552,89	50,53
			16384	471324,00	4713,24	4884,27	69,89	468184,00	4681,84	2152,54	46,40
	HASH	MD5	128	445816,00	4458,16	4694,76	68,52				
		SHA	160	453780,00	4537,80	3293,54	57,39				
			256	450644,00	4506,44	5312,94	72,89				
			512	448776,00	4487,76	4486,08	66,98				

Tabela A.3: Resultados referente ao tempo gasto (em segundos) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 100 MB para o Raspberry Pi.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
100 MB	AES	ECB	128	808,98	8,09	76,08	0,76	902,80	9,03	91,32	0,91
			192	871,90	8,72	86,69	9,31	795,58	7,96	72,58	8,52
			256	882,36	8,82	86,01	9,27	846,10	8,46	79,73	8,93
		CBC	128	789,12	7,89	70,72	8,41	880,24	8,80	89,23	9,45
			192	793,87	7,94	72,59	8,52	789,62	7,90	72,08	8,49
			256	746,37	7,46	63,07	7,94	778,38	7,78	69,15	8,32
		CTR	128	831,12	8,31	78,00	8,83	840,56	8,41	78,78	8,88
			192	828,16	8,28	78,10	8,84	890,69	8,91	90,44	9,51
			256	817,79	8,18	75,69	8,70	813,62	8,14	75,97	8,72
		OFB	128	892,04	8,92	90,10	9,49	901,41	9,01	91,88	9,59
			192	945,79	9,46	102,15	10,11	831,04	8,31	80,42	8,97
			256	901,65	9,02	89,58	9,46	930,87	9,31	93,58	9,67
		CFB	128	893,85	8,94	90,67	9,52	784,51	7,85	71,88	8,48
			192	881,29	8,81	89,55	9,46	934,38	9,34	97,23	9,86
			256	956,20	9,56	100,36	10,02	966,72	9,67	101,82	10,09
	DES	ECB	56	938,33	9,38	93,06	9,65	990,76	9,91	105,56	10,27
		CBC	56	985,87	9,86	103,95	10,20	1004,12	10,04	107,00	10,34
		OFB	56	993,52	9,94	104,73	10,23	975,65	9,76	100,44	10,02
		CFB	56	961,53	9,62	97,53	9,88	1004,81	10,05	107,68	10,38
	3DES	CBC	168	1636,73	16,37	272,54	16,51	1784,25	17,84	324,52	18,01
		OFB	168	1723,56	17,24	302,01	17,38	1772,91	17,73	319,69	17,88
		CFB	168	1727,01	17,27	304,94	17,46	1796,75	17,97	330,08	18,17
	RC4		128	843,88	8,44	85,54	9,25	875,49	8,75	90,27	9,50
	RSA		1024	144,86	1,45	2,10	1,45	143,94	1,44	2,07	1,44
			2048	146,46	1,46	2,15	1,46	143,98	1,44	2,07	1,44
			4096	157,89	1,58	2,49	1,58	143,97	1,44	2,07	1,44
			8192	236,88	2,37	5,61	2,37	144,46	1,44	2,09	1,44
			16384	828,15	8,28	68,58	8,28	147,06	1,47	2,16	1,47
	HASH	MD5	128	61,82	0,62	0,00025	0,01592				
		SHA1	160	79,22	0,79	0,00033	0,01829				
		SHA256	256	144,10	1,44	0,00058	0,02414				
		SHA512	512	150,73	1,51	0,00387	0,0622				

Tabela A.4: Resultados referente ao consumo de memória (em *Kilobytes*) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 100 MB para o Raspberry Pi.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s
100 MB	AES	ECB	128	259164	2591,64	3160,35	56,217	259604	2596,04	4122,08	64,2034
			192	257772	2577,72	2876,56	53,6336	259028	2590,28	3465,36	58,8673
			256	258448	2584,48	3812,57	61,746	258172	2581,72	4479,12	66,9262
		CBC	128	259704	2597,04	3430,28	58,5686	259224	2592,24	3214,02	56,6923
			192	260168	2601,68	3106,62	55,737	259628	2596,28	3287,76	57,339
			256	259612	2596,12	2636,95	51,3512	259896	2598,96	2812,04	53,0287
		CTR	128	260488	2604,88	3599,23	59,9935	260060	2600,6	260060	509,961
			192	259616	2596,16	3108,77	55,7564	259764	2597,64	3348,51	57,8663
			256	259464	2594,64	2564,55	50,6414	260584	2605,84	3188,61	56,4678
		OFB	128	260524	2605,24	3178,14	56,375	260336	2603,36	3193,19	56,5083
			192	259704	2597,04	2872,84	53,5989	259952	2599,52	3056,09	55,2819
			256	260848	2608,48	3483,29	59,0194	260408	2604,08	2741,75	52,3618
		CFB	128	259100	2591	2596,92	50,96	259512	2595,12	3161,47	56,2269
			192	259116	2591,16	2574,17	50,7363	259904	2599,04	2954,6	54,3562
			256	259472	2594,72	2593,24	50,9239	259820	2598,2	3259,32	57,0905
	DES	ECB	56	258648	2586,48	3775,29	37,7529	258100	2581	2745,08	52,3935
		CBC	56	258316	2583,16	3944,41	39,4441	258580	2585,8	4033,4	63,5091
		OFB	56	258288	2582,88	3335,71	33,3571	258324	2583,24	3233,18	56,8611
		CFB	56	257736	2577,36	3252,23	32,5223	259136	2591,36	3403,75	58,3417
	3DES	CBC	168	258332	2583,32	3540,18	35,4018	258684	2586,84	4183,77	64,6821
		OFB	168	259828	2598,28	5005,52	50,0552	259444	2594,44	3781,41	61,4931
		CFB	168	258396	2583,96	3886,56	38,8656	259372	2593,72	4050	63,6396
	RC4		128	257728	2577,28	2761,88	27,6188	258288	2582,88	2960,03	54,4061
	RSA		1024	259440	2594,40	7834,24	88,51	258048	2580,48	7570,65	87,01
			2048	262356	2623,56	8141,09	90,23	257952	2579,52	7063,77	84,05
			4096	262968	2629,68	8779,58	93,70	257116	2571,16	7632,73	87,37
			8192	266292	2662,92	8895,63	94,32	257704	2577,04	8088,20	89,93
			16384	270972	2709,72	8347,28	91,36	259672	2596,72	5992,76	77,41
	HASH	MD5	128	251592	2515,92	3085,57	55,55				
		SHA1	160	252316	2523,16	4181,75	64,67				
		SHA256	256	253664	2536,64	3319,51	57,62				
		SHA512	512	254228	2542,28	2399,44	48,98				

Tabela A.5: Resultados referente ao tempo gasto (em segundos) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 1 GB para o Computador Portátil.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
1 GB	AES	ECB	128	2467,26	24,67	35,00	5,92	1601,38	16,01	15,25	3,91
			192	2664,07	26,64	11,28	3,36	1229,39	12,29	1,72	1,31
			256	1631,16	16,31	5,46	2,34	1436,13	14,36	1,12	1,06
		CBC	128	2851,55	28,52	0,53	0,73	2350,28	23,50	1,71	1,31
			192	3302,17	33,02	3,10	1,76	3436,39	34,36	5,96	2,44
			256	3129,28	31,29	19,46	4,41	2838,06	28,38	88,36	9,40
		CTR	128	2810,13	28,10	0,99	1,00	1555,23	15,55	32,07	5,66
			192	3194,07	31,94	2,57	1,60	3291,67	32,92	5,68	2,38
			256	3252,86	32,53	16,17	4,02	3144,05	31,44	91,82	9,58
		OFB	128	2891,66	28,92	0,64	0,80	2819,59	28,20	76,95	8,77
			192	3585,15	35,85	3,99	2,00	3562,08	35,62	6,59	2,57
			256	3131,90	31,32	0,84	0,92	2473,37	24,73	1,01	1,01
		CFB	128	3137,49	31,37	0,78	0,88	2706,48	27,06	3,22	1,80
			192	3106,87	31,07	5,31	2,30	2480,89	24,81	2,96	1,72
			256	3141,68	31,42	2,90	1,70	2891,57	28,92	2,45	1,57
	DES	ECB	56	6037,24	60,37	11,60	3,41	5880,85	58,81	8,41	2,90
		CBC	56	5920,07	59,20	9,63	3,10	5777,89	57,78	7,70	2,78
		OFB	56	6338,13	63,38	8,90	2,98	6197,55	61,98	10,30	3,21
		CFB	56	6561,84	65,62	7,02	2,65	6434,43	64,34	13,04	3,61
	3DES	CBC	168	12854,02	128,54	4,47	2,11	12935,03	129,35	9,32	3,05
		OFB	168	13243,61	132,44	4,11	2,03	13306,85	133,07	8,08	2,84
		CFB	168	13512,32	135,12	8,39	2,90	13557,25	135,57	8,51	2,92
	RC4		128	1664,55	16,65	27,94	5,29	1662,27	16,62	35,34	5,94
	RSA		1024	1303,49	13,03	0,02	0,13	1302,27	13,02	0,00	0,02
			2048	1303,30	13,03	0,00	0,03	1301,89	13,02	0,00	0,02
			4096	1306,94	13,07	0,00	0,02	1302,21	13,02	0,00	0,02
			8192	1330,30	13,30	0,00	0,02	1302,29	13,02	0,00	0,03
			16384	1507,18	15,07	0,00	0,03	1303,03	13,03	0,00	0,02
	HASH	MD5	128	414,79	4,15	0,90	0,95				
		SHA	160	622,77	6,23	0,00	0,02				
			256	1297,67	12,98	0,00	0,02				
			512	1052,05	10,52	0,00	0,03				

Tabela A.6: Resultados referente ao consumo de memória (em *Kilobytes*) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 1 GB para o Computador Portátil.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s
1 GB	AES	ECB	128	431052,00	4310,52	8591,08	92,69	431196,00	4311,96	10565,65	102,79
			192	437828,00	4378,28	7173,90	84,70	437580,00	4375,80	5456,61	73,87
			256	434024,00	4340,24	8010,29	89,50	433036,00	4330,36	5771,34	75,97
		CBC	128	441852,00	4418,52	5265,02	72,56	442504,00	4425,04	5237,53	72,37
			192	441740,00	4417,40	5610,14	74,90	442664,00	4426,64	5886,78	76,73
			256	440492,00	4404,92	5950,34	77,14	442068,00	4420,68	7712,99	87,82
		CTR	128	441504,00	4415,04	5380,89	73,35	441060,00	4410,60	6750,51	82,16
			192	443916,00	4439,16	5351,21	73,15	443376,00	4433,76	5355,74	73,18
			256	442508,00	4425,08	5419,91	73,62	443036,00	4430,36	4832,35	69,52
		OFB	128	445072,00	4450,72	3714,87	60,95	438284,00	4382,84	4577,39	67,66
			192	437720,00	4377,20	3969,13	63,00	438556,00	4385,56	5094,75	71,38
			256	437916,00	4379,16	4534,40	67,34	438832,00	4388,32	4827,05	69,48
		CFB	128	439284,00	4392,84	4273,23	65,37	440196,00	4401,96	3234,42	56,87
			192	434580,00	4345,80	10629,62	103,10	435760,00	4357,60	6902,63	83,08
			256	438284,00	4382,84	4959,77	70,43	438084,00	4380,84	5279,77	72,66
	DES	ECB	56	438156,00	4381,56	10867,04	104,25	436796,00	4367,96	11994,67	109,52
		CBC	56	435896,00	4358,96	10885,37	104,33	436940,00	4369,40	11511,07	107,29
		OFB	56	437808,00	4378,08	12692,52	112,66	437584,00	4375,84	10022,12	100,11
		CFB	56	437472,00	4374,72	12198,10	110,45	437504,00	4375,04	12070,83	109,87
	3DES	CBC	168	434544,00	4345,44	5169,30	71,90	432228,00	4322,28	4765,50	69,03
		OFB	168	432136,00	4321,36	5542,21	74,45	432608,00	4326,08	4357,00	66,01
		CFB	168	434276,00	4342,76	4776,31	69,11	432784,00	4327,84	6060,90	77,85
	RC4		128	418916,00	4189,16	17623,37	132,75	419824,00	4198,24	14847,46	121,85
	RSA		1024	448668,00	4486,68	3609,23	60,08	455512,00	4555,12	3725,12	61,03
			2048	449272,00	4492,72	3668,49	60,57	455480,00	4554,80	3952,65	62,87
			4096	451808,00	4518,08	4402,90	66,35	454284,00	4542,84	4259,97	65,27
			8192	454056,00	4540,56	4775,76	69,11	456440,00	4564,40	3121,29	55,87
			16384	461620,00	4616,20	3775,15	61,44	458268,00	4582,68	3234,28	56,87
	HASH	MD5	128	431464,00	4314,64	3835,22	61,93				
		SHA	160	434240,00	4342,40	2592,97	50,92				
			256	434404,00	4344,04	3813,01	61,75				
			512	433928,00	4339,28	5736,53	75,74				

Tabela A.7: Resultados referente ao tempo gasto (em segundos) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 1 GB para o Raspberry Pi.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
1 GB	AES	ECB	128	13268,68	132,69	18,76	4,33	13565,16	135,65	15,11	3,89
			192	13218,52	132,19	10,37	3,22	13589,07	135,89	23,51	4,85
			256	13311,33	133,11	31,47	5,61	13584,00	135,84	16,83	4,10
		CBC	128	13238,24	132,38	12,62	3,55	13709,44	137,09	19,36	4,40
			192	13339,49	133,39	15,71	3,96	13695,60	136,96	15,50	3,94
			256	13352,38	133,52	12,96	3,60	13622,41	136,22	13,15	3,63
		CTR	128	13156,17	131,56	6,59	2,57	13402,20	134,02	13,15	3,63
			192	13175,23	131,75	7,95	2,82	13379,24	133,79	9,77	3,13
			256	13198,99	131,99	18,33	4,28	13431,21	134,31	10,73	3,28
		OFB	128	13695,29	136,95	25,95	5,09	13807,77	138,08	27,38	5,23
			192	13699,71	137,00	23,56	4,85	13870,51	138,71	21,69	4,66
			256	13704,05	137,04	31,37	5,60	13918,14	139,18	35,73	5,98
		CFB	128	13755,33	137,55	34,97	5,91	13834,72	138,35	19,68	4,44
			192	13618,88	136,19	14,80	3,85	13904,42	139,04	16,36	4,04
			256	13688,96	136,89	14,78	3,84	13987,86	139,88	27,54	5,25
	DES	ECB	56	14605,58	146,06	6,68	2,58	14600,53	146,01	15,14	3,89
		CBC	56	14036,01	140,36	8,70	2,95	14342,30	143,42	8,87	2,98
		OFB	56	14879,14	148,79	8,38	2,90	15182,48	151,82	13,40	3,66
		CFB	56	14847,56	148,48	12,51	3,54	15048,66	150,49	10,48	3,24
	3DES	CBC	168	21655,70	216,56	67,20	8,20	22569,88	225,70	5,58	2,36
		OFB	168	22409,19	224,09	73,37	8,57	23176,36	231,76	5,24	2,29
		CFB	168	22393,54	223,94	79,43	8,91	23356,21	233,56	9,05	3,01
	RC4		128	12671,29	126,71	35,05	5,92	13308,27	133,08	6,38	2,53
	RSA		1024	4374,23	43,74	5,67	2,38	4365,44	43,65	4,33	2,08
			2048	4366,32	43,66	4,38	2,09	4363,06	43,63	4,37	2,09
			4096	4398,95	43,99	7,01	2,65	4378,82	43,79	5,24	2,29
			8192	4460,85	44,61	4,22	2,06	4368,00	43,68	4,24	2,06
			16384	5056,16	50,56	4,27	2,07	4371,23	43,71	4,12	2,03
	HASH	MD5	128	4326,27	43,26	7,00	2,65				
		SHA	160	4347,01	43,47	6,50	2,55				
			256	4388,10	43,88	8,62	2,94				
			512	4398,90	43,99	8,42	2,90				

Tabela A.8: Resultados referente ao consumo de memória (em *Kilobytes*) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 1 GB para o Raspberry Pi.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s
1 GB	AES	ECB	128	258716,00	2587,16	4116,78	64,16	259068,00	2590,68	3875,57	62,25
			192	257236,00	2572,36	3041,32	55,15	258940,00	2589,40	4797,86	69,27
			256	257900,00	2579,00	4198,59	64,80	258884,00	2588,84	3957,75	62,91
		CBC	128	260052,00	2600,52	3303,32	57,47	259460,00	2594,60	3405,05	58,35
			192	259372,00	2593,72	2621,82	51,20	258936,00	2589,36	2661,24	51,59
			256	258908,00	2589,08	2966,58	54,47	259732,00	2597,32	2890,04	53,76
		CTR	128	258968,00	2589,68	2149,23	46,36	260208,00	2602,08	3105,12	55,72
			192	259572,00	2595,72	3618,02	60,15	260516,00	2605,16	3128,01	55,93
			256	259172,00	2591,72	3061,42	55,33	260244,00	2602,44	2766,51	52,60
		OFB	128	259840,00	2598,40	3041,29	55,15	259940,00	2599,40	3132,89	55,97
			192	259920,00	2599,20	3023,52	54,99	260288,00	2602,88	3017,08	54,93
			256	259836,00	2598,36	2828,31	53,18	260228,00	2602,28	3614,47	60,12
		CFB	128	259224,00	2592,24	2489,15	49,89	259424,00	2594,24	2802,85	52,94
			192	259860,00	2598,60	2862,67	53,50	259532,00	2595,32	3199,05	56,56
			256	259384,00	2593,84	3308,42	57,52	260888,00	2608,88	3709,28	60,90
	DES	ECB	56	258232,00	2582,32	4463,90	66,81	257972,00	2579,72	3710,14	60,91
		CBC	56	256544,00	2565,44	4311,12	65,66	257168,00	2571,68	4165,71	64,54
		OFB	56	257932,00	2579,32	3846,48	62,02	257028,00	2570,28	4293,25	65,52
		CFB	56	256692,00	2566,92	3593,00	59,94	257208,00	2572,08	3846,13	62,02
	3DES	CBC	168	258896,00	2588,96	4781,61	69,15	258416,00	2584,16	3789,23	61,56
		OFB	168	259224,00	2592,24	4820,31	69,43	259200,00	2592,00	4525,25	67,27
		CFB	168	259384,00	2593,84	4405,15	66,37	259140,00	2591,40	3977,82	63,07
	RC4		128	257244,00	2572,44	2437,46	49,37	257968,00	2579,68	3600,70	60,01
	RSA		1024	257780,00	2577,80	8575,15	92,60	254412,00	2544,12	5146,65	71,74
			2048	259368,00	2593,68	8791,33	93,76	256660,00	2566,60	6980,97	83,55
			4096	260776,00	2607,76	7259,42	85,20	256428,00	2564,28	7544,97	86,86
			8192	264052,00	2640,52	7473,67	86,45	255028,00	2550,28	6648,65	81,54
			16384	271072,00	2710,72	6747,44	82,14	259340,00	2593,40	6492,57	80,58
	HASH	MD5	128	250744,00	2507,44	2881,62	53,68				
		SHA	160	251360,00	2513,60	3705,86	60,88				
			256	253012,00	2530,12	3016,55	54,92				
			512	252804,00	2528,04	2639,35	51,37				

Tabela A.9: Resultados referente ao tempo gasto (em segundos) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 2 GB para o Computador Portátil.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
2 GB	AES	ECB	128	7175,58	71,76	10,00	3,16	6175,60	61,76	54,27	7,37
			192	7409,41	74,09	4,70	2,17	5944,90	59,45	44,80	6,69
			256	6974,87	69,75	32,34	5,69	5320,20	53,20	218,58	14,78
		CBC	128	7317,14	73,17	3,04	1,74	5037,70	50,38	14,18	3,77
			192	7679,56	76,80	17,88	4,23	6541,27	65,41	193,50	13,91
			256	7812,67	78,13	11,57	3,40	6754,36	67,54	121,88	11,04
		CTR	128	7732,50	77,33	11,51	3,39	6784,54	67,85	84,06	9,17
			192	7400,92	74,01	5,11	2,26	4974,28	49,74	8,29	2,88
			256	7112,36	71,12	8,17	2,86	5917,85	59,18	37,88	6,15
		OFB	128	7484,16	74,84	34,33	5,86	6652,69	66,53	89,05	9,44
			192	6653,86	66,54	1,93	1,39	4713,53	47,14	7,11	2,67
			256	6757,42	67,57	17,54	4,19	4904,96	49,05	110,32	10,50
		CFB	128	6806,08	68,06	8,22	2,87	5015,99	50,16	12,45	3,53
			192	7314,37	73,14	90,72	9,52	5952,19	59,52	328,64	18,13
			256	7253,54	72,54	25,73	5,07	6474,91	64,75	174,13	13,20
	DES	ECB	56	12354,19	123,54	46,46	6,82	12111,55	121,12	68,62	8,28
		CBC	56	12182,80	121,83	39,61	6,29	11850,78	118,51	49,60	7,04
		OFB	56	13054,48	130,54	202,46	14,23	12765,86	127,66	54,49	7,38
		CFB	56	13495,09	134,95	473,99	21,77	13413,03	134,13	323,47	17,99
	3DES	CBC	168	26141,33	261,41	389,99	19,75	25777,64	257,78	76,89	8,77
		OFB	168	26761,68	267,62	59,05	7,68	26534,22	265,34	53,88	7,34
		CFB	168	27207,42	272,07	53,11	7,29	27040,75	270,41	54,21	7,36
	RC4		128	6868,18	68,68	21,41	4,63	6021,18	60,21	131,47	11,47
	RSA		1024	2623,83	26,24	2,65	1,63	2601,97	26,02	0,00379	0,0616
			2048	2631,58	26,32	8,81	2,97	2608,84	26,09	0,5409	0,73546
			4096	2607,50	26,08	0,02646	0,16267	2602,85	26,03	0,00593	0,07698
			8192	2642,49	26,42	0,78709	0,88718	2612,35	26,12	0,53986	0,73475
			16384	2852,47	28,52	3,13121	1,76952	2634,22	26,34	2,75381	1,65946
	HASH	MD5	128	3725,75	37,26	11,20	3,35				
		SHA	160	3782,14	37,82	8,30	2,88				
			256	3780,69	37,81	12,84	3,58				
			512	3453,67	34,54	1,12	1,06				

Tabela A.10: Resultados referente ao consumo de memória (em *Kilobytes*) dos algoritmos DES, 3DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 2 GB para o Computador Portátil.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
2 GB	AES	ECB	128	440388,00	4403,88	3689,84	60,74	440780,00	4407,80	3118,02	55,84
			192	440636,00	4406,36	2432,03	49,32	441388,00	4413,88	2039,74	45,16
			256	439932,00	4399,32	3848,10	62,03	440512,00	4405,12	3025,48	55,00
		CBC	128	425736,00	4257,36	3989,73	63,16	424552,00	4245,52	4222,64	64,98
			192	426432,00	4264,32	4309,55	65,65	427596,00	4275,96	4266,83	65,32
			256	424028,00	4240,28	5275,56	72,63	424524,00	4245,24	5018,08	70,84
		CTR	128	425644,00	4256,44	3959,24	62,92	424856,00	4248,56	4268,29	65,33
			192	424944,00	4249,44	4697,06	68,54	425492,00	4254,92	5275,75	72,63
			256	437792,00	4377,92	4148,52	64,41	438912,00	4389,12	3727,54	61,05
		OFB	128	431080,00	4310,80	5264,32	72,56	432228,00	4322,28	6368,73	79,80
			192	429236,00	4292,36	3608,27	60,07	429576,00	4295,76	3331,82	57,72
			256	444508,00	4445,08	3437,21	58,63	442856,00	4428,56	3119,20	55,85
		CFB	128	436516,00	4365,16	3387,57	58,20	435200,00	4352,00	2777,54	52,70
			192	435860,00	4358,60	2712,04	52,08	435932,00	4359,32	3295,69	57,41
			256	434048,00	4340,48	2178,68	46,68	434952,00	4349,52	2271,93	47,66
	DES	ECB	56	433144,00	4331,44	5648,17	75,15	435268,00	4352,68	5849,23	76,48
		CBC	56	433840,00	4338,40	5280,00	72,66	434024,00	4340,24	4885,92	69,90
		OFB	56	435048,00	4350,48	4435,65	66,60	433364,00	4333,64	5433,24	73,71
		CFB	56	433760,00	4337,60	6624,97	81,39	434872,00	4348,72	4749,38	68,92
	3DES	CBC	168	434000,00	4340,00	4541,09	67,39	433324,00	4333,24	4765,96	69,04
		OFB	168	433420,00	4334,20	4346,30	65,93	434044,00	4340,44	4050,39	63,64
		CFB	168	434276,00	4342,76	4632,47	68,06	433260,00	4332,60	4979,84	70,57
	RC4		128	433788,00	4337,88	4744,23	68,88	433252,00	4332,52	4679,00	68,40
	RSA		1024	445424,00	4454,24	4579,01	67,67	449092,00	4490,92	4056,84	63,69
			2048	444604,00	4446,04	2529,13	50,29	450120,00	4501,20	3125,17	55,90
			4096	447236,00	4472,36	4283,51	65,45	450352,00	4503,52	2803,48	52,95
			8192	448744,00	4487,44	3294,39	57,40	451408,00	4514,08	2956,76	54,38
			16384	456224,00	4562,24	3362,04	57,98	453108,00	4531,08	2750,66	52,45
	HASH	MD5	128	399768,00	3997,68	27672,95	166,35				
		SHA	160	392348,00	3923,48	16276,25	127,58				
			256	439424,00	4394,24	5193,80	72,07				
			512	425780,00	4257,80	3180,08	56,39				

Tabela A.11: Resultados referente ao tempo gasto (em segundos) dos algoritmos DES, RC4, AES, RSA,MD5 e SHA com ficheiro de 2 GB para o Raspberry Pi.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} t_i$	$\bar{x}$	s^2	s
2 GB	AES	ECB	128	19710,69	197,11	153,74	12,40	20908,62	209,09	393,24	19,83
			192	20073,71	200,74	291,63	17,08	20822,85	208,23	316,71	17,80
			256	20071,41	200,71	352,22	18,77	20576,92	205,77	159,44	12,63
		CBC	128	19535,56	195,36	161,55	12,71	20922,68	209,23	564,96	23,77
			192	19948,18	199,48	395,64	19,89	20647,52	206,48	333,27	18,26
			256	19771,55	197,72	422,00	20,54	20487,53	204,88	186,33	13,65
		CTR	128	20008,34	200,08	606,76	24,63	20368,03	203,68	263,77	16,24
			192	19725,94	197,26	338,54	18,40	20538,99	205,39	265,23	16,29
			256	19779,52	197,80	326,20	18,06	20592,96	205,93	280,22	16,74
		OFB	128	19745,04	197,45	351,91	18,76	20558,06	205,58	210,17	14,50
			192	19691,45	196,91	204,20	14,29	20726,27	207,26	384,95	19,62
			256	20117,30	201,17	348,49	18,67	20812,09	208,12	327,93	18,11
		CFB	128	19871,52	198,72	321,81	17,94	20644,32	206,44	212,68	14,58
			192	19569,61	195,70	126,29	11,24	20850,83	208,51	353,84	18,81
			256	20084,54	200,85	360,21	18,98	20807,01	208,07	325,56	18,04
	DES	ECB	56	22614,63	226,15	133,10	11,54	22621,27	226,21	355,20	18,85
		CBC	56	21853,93	218,54	244,87	15,65	22193,79	221,94	293,40	17,13
		OFB	56	23451,97	234,52	421,41	20,53	23417,35	234,17	143,52	11,98
		CFB	56	24081,75	240,82	458,37	21,41	23621,67	236,22	221,14	14,87
	RC4		128	20403,68	204,04	500,93	22,38	20780,44	207,80	514,27	22,68
	RSA		1024	9827,58	98,28	6,16	2,48	9849,46	98,49	11,15	3,34
			2048	9853,24	98,53	12,84	3,58	9822,22	98,22	4,37	2,09
			4096	9834,79	98,35	3,97	1,99	9825,90	98,26	4,13	2,03
			8192	9899,99	99,00	0,70594	0,84	9814,27	98,14	1,22318	1,11
			16384	10600,16	106,00	16,26	4,03	9841,58	98,42	7,62	2,76
	HASH	MD5	128	9910,76	99,11	1,55	1,25				
		SHA	160	9920,62	99,21	2,18	1,48				
			256	9797,86	97,98	2,03	1,42				
			512	9790,42	97,90	0,24	0,49				

Tabela A.12: Resultados referente ao consumo de memória (em *Kilobytes*) dos algoritmos DES, RC4, AES, RSA, MD5 e SHA com ficheiro de 2 GB para o Raspberry Pi.

Ficheiro	Algoritmo de Cifra			Cifrar/Assinar				Decifrar/Verificar			
	Cifras	Modos	Chave/bits	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s	$\sum_{i=1}^{100} m_i$	$\bar{x}$	s^2	s
2 GB	AES	ECB	128	257876,00	2578,76	2927,74	54,11	259076,00	2590,76	3601,68	60,01
			192	258804,00	2588,04	3221,82	56,76	259972,00	2599,72	3991,68	63,18
			256	259988,00	2599,88	4128,79	64,26	257644,00	2576,44	3695,48	60,79
		CBC	128	258644,00	2586,44	2541,54	50,41	259924,00	2599,24	3796,59	61,62
			192	259596,00	2595,96	3206,30	56,62	259840,00	2598,40	3590,79	59,92
			256	260528,00	2605,28	3127,88	55,93	259836,00	2598,36	3737,57	61,14
		CTR	128	260388,00	2603,88	2209,12	47,00	260760,00	2607,60	3614,87	60,12
			192	259968,00	2599,68	3162,08	56,23	260344,00	2603,44	2988,29	54,67
			256	260124,00	2601,24	2928,39	54,11	259900,00	2599,00	2981,29	54,60
		OFB	128	260588,00	2605,88	3325,88	57,67	261020,00	2610,20	2942,51	54,24
			192	260264,00	2602,64	3599,91	60,00	260968,00	2609,68	3239,82	56,92
			256	260944,00	2609,44	3950,71	62,85	259196,00	2591,96	3150,38	56,13
		CFB	128	261844,00	2618,44	3584,29	59,87	259132,00	2591,32	2651,49	51,49
			192	260676,00	2606,76	3278,45	57,26	260052,00	2600,52	2540,17	50,40
			256	259208,00	2592,08	2593,93	50,93	260884,00	2608,84	3356,54	57,94
	DES	ECB	56	257548,00	2575,48	3353,10	57,91	258044,00	2580,44	2975,97	54,55
		CBC	56	259460,00	2594,60	4663,07	68,29	258952,00	2589,52	3903,61	62,48
		OFB	56	258164,00	2581,64	3134,41	55,99	259032,00	2590,32	3331,94	57,72
		CFB	56	258452,00	2584,52	3149,79	56,12	258424,00	2584,24	3342,81	57,82
	RC4		128	258620,00	2586,20	3061,45	55,33	258164,00	2581,64	2785,32	52,78
	RSA		1024	259744,00	2597,44	7601,30	87,19	254672,00	2546,72	6798,51	82,45
			2048	259472,00	2594,72	8554,95	92,49	256392,00	2563,92	8577,61	92,62
			4096	262616,00	2626,16	8124,90	90,14	255968,00	2559,68	6961,03	83,43
			8192	265004,00	2650,04	9888,81	99,44	257120,00	2571,20	6812,12	82,54
			16384	270864,00	2708,64	9801,93	99,00	259756,00	2597,56	7628,90	87,34
	HASH	MD5	128	251100,00	2511,00	2954,46	54,35				
		SHA	160	253524,00	2535,24	3743,90	61,19				
			256	254228,00	2542,28	3286,06	57,32				
			512	255220,00	2552,20	3276,73	57,24				