Dissertation Type: software development

DEPARTMENT OF COMPUTER SCIENCE

Building a Testbed for Evaluating Privacy Enhancing Technologies (PETs)

Jacob Daniel Halsey

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of Bachelor of Science in the Faculty of Engineering.

Friday 14th May, 2021

Declaration

This dissertation is submitted to the University of Bristol in accordance with the requirements of the degree of BSc in the Faculty of Engineering. It has not been submitted for any other degree or diploma of any examining body. Except where specifically acknowledged, it is all the work of the Author.

Jacob Daniel Halsey, Friday 14$^{\text{th}}$ May, 2021

Contents

1 Contextual Background **1**

1.1 What are Privacy Enhancing Technologies (PETs)? 1

1.2 Existing Solutions 2

1.3 High Level Objectives 2

2 Technical Background **5**

2.1 Virtualization 5

2.2 Containerization 6

2.3 Virtual Networks 6

2.4 Software Defined Networking 6

2.5 The Rust Language 6

3 Project Execution **7**

3.1 Example Section 7

4 Critical Evaluation **13**

5 Conclusion **15**

Executive Summary

The goal of this project is to produce a simple and lightweight testbed platform for evaluating privacy enhancing technologies. It should provide support for testing varied architectures and network topologies, such as client-server and peer-to-peer applications. It should also support simulating applications for different types of platforms including mobile phone apps.

Summary of work:

- I have developed a flexible command line tool called *kvm-compose* for Linux using the Rust language and *libvirt* library for building and destroying virtual testing environments.
- In the process I have made some contributions to the *libvirt-rust* language bindings open source library.
- I have then implemented some example projects using the testbed tool.

Supporting Technologies

- *Linux KVM* (Kernel-based Virtual Machine) - https://www.linux-kvm.org/
- *Open vSwitch* Virtual multilayer switch - https://www.openvswitch.org/
- *libvirt* Virtualization API - https://libvirt.org/
- *Rust* Language, Compiler, Toolchain, etc. - https://www.rust-lang.org/
- *libvirt-rust* Rust bindings to the libvirt - https://gitlab.com/libvirt/libvirt-rust
- *clap* Rust command Line Argument Parser - https://github.com/clap-rs/clap
- *serde* Rust Serialization framework - https://github.com/serde-rs/
- *serde-yaml* YAML backend for serde - https://github.com/dtolnay/serde-yaml
- *serde-plain* Plain text backend for serde - https://github.com/mitsuhiko/serde-plain
- *thiserror* Rust error derive macro - https://github.com/dtolnay/thiserror
- *anyhow* Rust error handling framework - https://github.com/dtolnay/anyhow
- *simple_logger* Rust logging implementation - https://github.com/borntyping/rust-simple_logger
- *xml-rs* XML library for Rust - https://github.com/netvl/xml-rs
- *validator* Rust struct validation - https://github.com/Keats/validator
- *directories* User data directories library - https://github.com/dirs-dev/directories-rs
- *reqwest* Rust HTTP Client - https://github.com/seanmonstar/reqwest
- *indicatif* Rust command line progress indicator - https://github.com/mitsuhiko/indicatif
- *tempfile* Rust temporary file library - https://github.com/Stebalien/tempfile
- *casual* Rust user input parser - https://github.com/rossmacarthur/casual
- *derive-new* Rust new constructor macro - https://github.com/nrc/derive-new
- *enum-iterator* Rust macro for iterating enums - https://github.com/stephaneyfx/enum-iterator
- *rust-embed* Embeds files into Rust binaries - https://github.com/pyros2097/rust-embed

Acknowledgements

I would like to thank my supervisor Professor Awais Rashid and co-supervisor Joe Gardiner for their project proposal and support and guidance in completing it.

Chapter 1

Contextual Background

The UK Research and Innovation (UKRI) is a non-departmental public body of the United Kingdom Government sponsored by the Department for Business, Energy and Industrial Strategy [1]. In October 2020 the UKRI announced the creation of the National Research Centre on Privacy, Harm Reduction and Adversarial Influence Online (REPHRAIN) [2]. The centre is made up of researchers in computer science, international relations, law, psychology, management, design, digital humanities, public policy, political Science, criminology, and sociology from five British universities including the University of Bristol.

REPHRAIN should be understood in the context of the UK government's *Online Harms White Paper* public consultation beginning in April 2019 [3], which sets out plans for new online safety measures; REPHRAIN's missions and outcomes are aligned with this paper [4].

REPHRAIN will focus on three core missions [5]:

1. Delivering privacy at scale while mitigating its misuse to inflict harms.
2. Minimising harms while maximising benefits from a sharing-driven digital economy.
3. Balancing individual agency vs. the social good.

The three missions will require looking at Privacy Enhancing Technologies (PETs); including their capabilities, applications of PETs in addressing existing online harms, mitigating the potential abuse of PETs, embedding the PETs into infrastructures, and developing new PETs. In order to facilitate this REPHRAIN intends to build a toolbox of resources including a PETs testbed. The testbed will be used researchers in developing, testing, and evaluating the PETs. The aim of this project is to develop a prototype for this testbed.

1.1 What are Privacy Enhancing Technologies (PETs)?

Before we discuss Privacy Enhancing Technologies we must consider what we mean by "privacy". REPHRAIN is primarily using the definitions set out by D. J. Solove in his 2006 article *A Taxonomy of Privacy* [6, 4]. Solove notes that the definition of privacy has often been very

1. Information collection
   (a) Surveillance
   (b) Interrogation
2. Information processing
   (a) Aggregation
   (b) Identification
   (c) Insecurity
   (d) Secondary Use
   (e) Exclusion
3. Information dissemination
   (a) Breach of Confidentiality
   (b) Disclosure
   (c) Exposure
   (d) Increased Accessibility
   (e) Blackmail
   (f) Appropriation
   (g) Distortion
4. Invasion
   (a) Intrusion
   (b) Decisional Interference

Figure 1.1: A Taxonomy of Privacy Violations [6].

broad or vague, and therefore sets out to develop a taxonomy of privacy violations. He has defined four groups of harmful activities (See figure 1.1).

Broadly speaking a Privacy Enhancing Technology is any solution or approach in hardware or software that helps protect a user from such privacy violations [7]. Some examples of PETs could include Onion routing such as the Tor network (which enables anonymous communication), or end-to-end encrypted messaging systems such as the Signal protocol. Kaaniche et al. [8] have defined a more comprehensive classification of PETs (see Figure 1.2).

1.2 Existing Solutions

There has been some existing research by Tekeoglu and Tosun [9] who have developed a privacy testbed for Internet-of-Things (IoT) devices. Their approach has some similar goals to this project in that it looks at capturing layer 2 and 3 network traffic. They note that the testbed enables experiments such as port vulnerability scans, checking what cipher suites are used (or not), and generally monitoring network traffic to see what data is being collected. However their testbed is different in that it is only designed for IoT devices; rather than general purpose PET applications.

1.3 High Level Objectives

Overall the high-level objective of this project is to develop a simple and lightweight testbed platform for evaluating PETs:

- The testbed should support testing various architectures and network topologies, including client/server and peer-to-peer applications, to accommodate a variety of PETs.

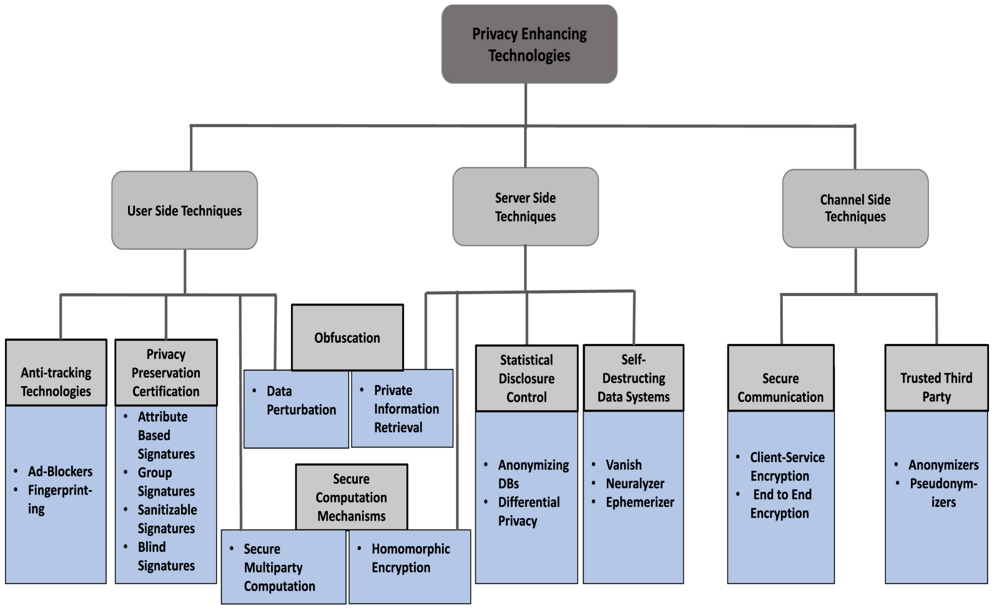

Figure 1.2: A Taxonomy of privacy enhancing technologies [8].

- The testbed must be able to collect information such as packet captures for use in evaluating the privacy properties.
- The testbed should support different platforms such as desktop and mobile apps, and both applications where the source code is available or only pre-built binaries.
- The testbed should enable a high level of automation, such that working with large test environments because feasible, and the setup can easily and programmatically be replicated.

Chapter 2

Technical Background

In this chapter I will discuss some of the technologies which this project depends or builds upon.

2.1 Virtualization

> 'Virtualization uses software to create an abstraction layer over computer hardware that allows the hardware elements of a single computer—processors, memory, storage and more—to be divided into multiple virtual computers, commonly called virtual machines (VMs). Each VM runs its own operating system (OS) and behaves like an independent computer, even though it is running on just a portion of the actual underlying computer hardware.' [10]

Virtualization will therefore be a very useful technology for the testbed, since it will allow us to model an environment consisting of multiple computers such as application clients and servers, and run them all within a single machine. In addition, modern CPU extensions (such as Intel VT and AMD-V) provide hardware-assisted / accelerated virtualization support, allowing the virtual machines to have near native performance which will help in meeting the goal of minimal overhead for the testbed.

In order to use virtualization a *Hypervisor* is required, this is the software layer sits between the physical hardware and manages the virtual machines. A hypervisor may run directly on the physical machine in place of a conventional operating system - a Type 1 or *bare-metal* hypervisor, or run within a separate host operating system - a Type 2 or *hosted* hypervisor.

2.1.1 KVM

For the prototype testbed I will be using *Kernel-based Virtual Machine* (KVM), which is a kernel module for the Linux operating system that allows it to function as a hypervisor. The advantage is that KVM (and the Linux kernel itself) are free and open-source software under

GNU licenses, and it is a stable and mature platform [11]. In userspace QEMU [1] may then use KVM to provide a full virtualization platform.

libvirt is an open-source toolkit for managing virtualization platforms [12], it supports QEMU/KVM as well as hypervisors from other vendors. It is written in C, with bindings available in many other programming languages, making it a suitable library for developing the testbed. Support for other platforms also means it would be easier to add support for additional hypervisors in future.

2.2 Containerization

2.3 Virtual Networks

2.4 Software Defined Networking

2.5 The Rust Language

A compulsory chapter, of roughly 10 pages

This chapter is intended to describe the technical basis on which execution of the project depends. The goal is to provide a detailed explanation of the specific problem at hand, and existing work that is relevant (e.g., an existing algorithm that you use, alternative solutions proposed, supporting technologies).

Per the same advice in the handbook, note there is a subtly difference from this and a full-blown literature review (or survey). The latter might try to capture and organise (e.g., categorise somehow) *all* related work, potentially offering meta-analysis, whereas here the goal is simple to ensure the dissertation is self-contained. Put another way, after reading this chapter a non-expert reader should have obtained enough background to understand what *you* have done (by reading subsequent sections), then accurately assess your work. You might view an additional goal as giving the reader confidence that you are able to absorb, understand and clearly communicate highly technical material.

[1] Note QEMU can also function as its own independent type 2 hypervisor but KVM is required to enable hardware acceleration, see `https://www.packetflow.co.uk/what-is-the-difference-between-qemu-and-kvm/`

Chapter 3

Project Execution

A topic-specific chapter, of roughly 15 pages

This chapter is intended to describe what you did: the goal is to explain the main activity or activities, of any type, which constituted your work during the project. The content is highly topic-specific, but for many projects it will make sense to split the chapter into two sections: one will discuss the design of something (e.g., some hardware or software, or an algorithm, or experiment), including any rationale or decisions made, and the other will discuss how this design was realised via some form of implementation.

This is, of course, far from ideal for *many* project topics. Some situations which clearly require a different approach include:

- In a project where asymptotic analysis of some algorithm is the goal, there is no real “design and implementation” in a traditional sense even though the activity of analysis is clearly within the remit of this chapter.
- In a project where analysis of some results is as major, or a more major goal than the implementation that produced them, it might be sensible to merge this chapter with the next one: the main activity is such that discussion of the results cannot be viewed separately.

Note that it is common to include evidence of “best practice” project management (e.g., use of version control, choice of programming language and so on). Rather than simply a rote list, make sure any such content is useful and/or informative in some way: for example, if there was a decision to be made then explain the trade-offs and implications involved.

3.1 Example Section

This is an example section; the following content is auto-generated dummy text. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

foo

Figure 3.1: This is an example figure.

foo	bar	baz
0	0	0
1	1	1
⋮	⋮	⋮
9	9	9

Table 3.1: This is an example table.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

3.1.1 Example Sub-section

This is an example sub-section; the following content is auto-generated dummy text. Notice the examples in Figure 3.1, Table 3.1, Algorithm 3.1 and Listing 3.1. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla

for $i = 0$ **upto** n **do**
  $t_i \leftarrow 0$
end

Algorithm 3.1: This is an example algorithm.

```
for( i = 0; i < n; i++ ) {
  t[ i ] = 0;
}
```

Listing 3.1: This is an example listing.

vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Example Sub-sub-section

This is an example sub-sub-section; the following content is auto-generated dummy text. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu.

Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Example paragraph. This is an example paragraph; note the trailing full-stop in the title, which is intended to ensure it does not run into the text.

Chapter 4

Critical Evaluation

A topic-specific chapter, of roughly 15 pages

This chapter is intended to evaluate what you did. The content is highly topic-specific, but for many projects will have flavours of the following:

1. functional testing, including analysis and explanation of failure cases,
2. behavioural testing, often including analysis of any results that draw some form of conclusion wrt. the aims and objectives, and
3. evaluation of options and decisions within the project, and/or a comparison with alternatives.

This chapter often acts to differentiate project quality: even if the work completed is of a high technical quality, critical yet objective evaluation and comparison of the outcomes is crucial. In essence, the reader wants to learn something, so the worst examples amount to simple statements of fact (e.g., "graph X shows the result is Y"); the best examples are analytical and exploratory (e.g., "graph X shows the result is Y, which means Z; this contradicts [1], which may be because I use a different assumption"). As such, both positive *and* negative outcomes are valid *if* presented in a suitable manner.

Chapter 5

Conclusion

A compulsory chapter, of roughly 5 pages

The concluding chapter of a dissertation is often underutilised because it is too often left too close to the deadline: it is important to allocation enough attention. Ideally, the chapter will consist of three parts:

1. (Re)summarise the main contributions and achievements, in essence summing up the content.

2. Clearly state the current project status (e.g., "X is working, Y is not") and evaluate what has been achieved with respect to the initial aims and objectives (e.g., "I completed aim X outlined previously, the evidence for this is within Chapter Y"). There is no problem including aims which were not completed, but it is important to evaluate and/or justify why this is the case.

3. Outline any open problems or future plans. Rather than treat this only as an exercise in what you *could* have done given more time, try to focus on any unexplored options or interesting outcomes (e.g., "my experiment for X gave counter-intuitive results, this could be because Y and would form an interesting area for further study" or "users found feature Z of my software difficult to use, which is obvious in hindsight but not during at design stage; to resolve this, I could clearly apply the technique of Smith [7]").

Bibliography

[1] UKRI, *Who we are*, Nov. 2020. [Online]. Available: https://www.ukri.org/about-us/who-we-are/.

[2] ——, *New centre launched to keep citizens safe online*, Oct. 2020. [Online]. Available: https://www.ukri.org/news/new-centre-launched-to-keep-citizens-safe-online/.

[3] Department for Digital, Culture, Media and Sport, *Online harms white paper*, Dec. 2020. [Online]. Available: https://www.gov.uk/government/consultations/online-harms-white-paper.

[4] REPHRAIN, *Online harms*. [Online]. Available: https://www.rephrain.ac.uk/online-harms/.

[5] ——, *Missions*. [Online]. Available: https://www.rephrain.ac.uk/missions/.

[6] D. J. Solove, "A taxonomy of privacy," *University of Pennsylvania Law Review*, vol. 154, no. 3, pp. 477–564, 2006. [Online]. Available: https://www.law.upenn.edu/journals/lawreview/articles/volume154/issue3/Solove154U.Pa.L.Rev.477%282006%29.pdf.

[7] D. Buckley, *Privacy enhancing technologies for trustworthy use of data*, Feb. 2021. [Online]. Available: https://cdei.blog.gov.uk/2021/02/09/privacy-enhancing-technologies-for-trustworthy-use-of-data/.

[8] N. Kaaniche, M. Laurent, and S. Belguith, "Privacy enhancing technologies for solving the privacy-personalization paradox: Taxonomy and survey," *Journal of Network and Computer Applications*, vol. 171, p. 102 807, 2020, ISSN: 1084-8045. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804520302794.

[9] A. Tekeoglu and A. S. Tosun, "A testbed for security and privacy analysis of iot devices," in *2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, IEEE, 2016, pp. 343–348. [Online]. Available: https://ieeexplore.ieee.org/document/7815045.

[10] IBM Cloud Education, *Virtualization: A complete guide*, Jun. 2019. [Online]. Available: https://www.ibm.com/cloud/learn/virtualization-a-complete-guide.

[11] Red Hat, *Kvm vs. vmware*. [Online]. Available: https://www.redhat.com/en/topics/virtualization/kvm-vs-vmware-comparison.

[12] libvirt, *The virtualization api*. [Online]. Available: https://libvirt.org/index.html.