Strengthening Capsicum Capabilities with Libpreopen

(Changed the title by modifying the file `thesis.tex`*)*

by

© *Stanley Uche Godfrey* (change this in `thesis.tex`)

A thesis submitted to the

School of Graduate Studies

in partial fulfilment of the

requirements for the degree of

Master of *Science* (change this in `thesis.tex`)

Department of *Scientific Computing* (change this in `thesis.tex`)

Memorial University of Newfoundland

December 2017 (change this in `thesis.tex`, too)

St. John's Newfoundland

Abstract

This document provides information on how to write your thesis using the LaTeX document preparation system. You can use these files as a template for your own thesis, just replace the content, as necessary. You should put your real abstract here, of course.

"The purpose of the abstract, which should not exceed 150 words for a Masters' thesis or 350 words for a Doctoral thesis, is to provide sufficient information to allow potential readers to decide on relevance of the thesis. Abstracts listed in Dissertation Abstracts International or Masters' Abstracts International should contain appropriate key words and phrases designed to assist electronic searches."

— MUN School of Graduate Studies

Acknowledgements

Put your acknowledgements here...

"Intellectual and practical assistance, advice, encouragement and sources of monetary support should be acknowledged. It is appropriate to acknowledge the prior publication of any material included in the thesis either in this section or in the introductory chapter of the thesis."

— MUN School of Graduate Studies

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Gargets to be Secured

As more networks and gadgets are connected to the internet, the web becomes indispensable, it hosts productivity software suites for creating documents, spreadsheets, and emails.Applications suites for making scientifi

c calculations, live television streaming and weather details hosted on the web.Some of the services web applications provide are online banking services, services for storing pictures and documents in the cloud,personal computers and mobile devices.Web applications also provide services that connect home devices such IP cameras to mobile phones for remote monitoring and e-commerce services. Sensitive data like passwords, credit card details are usually required to access these web services. These web applications that provide web services can have some vulnerabilities which attackers can exploit, despite network defenses like firewall and intrusion prevention systems [3].

One such vulnerability in an application could be as a result of a Buffer Overflow [4].A Buffer Overflow is a programming bug usually in C and C++ programs that resulted when a programmer fails to check if an input data is within the bounds of that input buffer. If the input data is more than the buffer can accommodate, the overflowing data will overwrite the content of the buffer or are written in adjacent memory and are pushed into the stack as an instruction to be executed. Buffer overflow attack is not that easy to carry out, in most cases the attacker spends hours studying the flow of execution of the program to be attacked using a debugger program such as OllyDgb . An attacker who is able to add more data in a buffer than the buffer can accommodate could change execution path of applications intentionally and may acquire the root user right of the system which will allow the attacker to take total control of the system.

If a web application is vulnerable, attackers can use a technique known as Heap Spray [1] to send malicious code to the heap memory of the web application in a computer. The Heap Spray technique is used to duplicate the malicious code in different locations of the running application's heap memory to increase the chances of execution of the malicious code. Heap spray is created with scripting languages like JavaScript. Different Malware exploited vulnerabilities found in internet explorer 6 and 7 around 2004 when the internet explorer web browser was believed to be the most popular web browser. Some of the vulnerabilities exploited in internet explorer include.ANI(CVE2007-0038),VML(CVE-2006-4868) and Operation Aurora exploit (CVE-2010-0248).

The first known buffer overflow [4] exploitation that gained mainstream media attention was accomplished by a graduate student of Cornell University know as Robert Tappan Morris. Moris wrote an experimental program that duplicates itself in a computer and disseminates itself to other computers through a computer network. Moris was able to put this program on the internet which was fast replicating,infecting and refecting computer at a fast rate.

Moris' program known as worm exploited buffer overflow bug in UNIX Sendmail program, a program which runs on a computer and waits for connections from other computers which it receives emails from. Moris program also exploited the buffer overflow in the finger, a daemon finger which servers as finger request.

These sorts of unauthorized access to computer resources by attackers are what Capsicum mitigates, and Libpreopen fortifies Capsicum in limiting the damage intrusive malicious code from such cyber attack could cause.

1.2 Background

Capsicum is a system that boosts UNIX security with sandboxed capability mode and capabilities. Capability mode is the ability of Capsicum to split an application into fragments that can interact with each other only in a regulated manner using Capsicum capabilities. Fragments of application in capability mode are totally isolated and these fragments are not allowed access to shared resources, make system calls, fork themselves or access global namespaces. The reason for these restrictions

is to in worst case scenario, contain vulnerabilities to a fragment or a process should one get corrupted, and not allow the corruption to spread to other fragments or the entire system.

. .

However, processes in total isolation cannot perform any task, this is where Capsicum capabilities are made use of. Capsicum capabilities are used to grant isolated processes in Capsicum capability mode regulated rights to perform specified actions in the capability token on a shared resources. For instance, a process may inherit file descriptor from a parent process or may request access to a system file from another process that has the right to send the file descriptor of the requested system file through IPC and before each of the processes enters Capsicum capability mode. Regardless of how capability rights are acquired. Processes in capability mode can only perform actions allowed in the capabilities granted on the file descriptors they acquire. A file descriptor acquired with capability right of CAP_READ cannot have fchmod(2) or CAP_WRITE operation perform on it. [2]

For an application to be compartmentalized by Capsicum, the application developer would make some modification to make the application conform to Capsicum features. The modifications could be rigorous and time-consuming. These difficult application modifications are what Libpreopen relieves application developers who want to make use of Capsicum compartmentalization features of.

Chapter 2

Design and Implementation of Libpreopen

2.1 Design

The design of Libpreopen was made to strengthen Capsicum from these two viewpoints.

(1) Libpreopen fortifies Capsicum by making it possible for an application running in Capsicum capability mode to run some commands that require System calls and access global namespaces without compromising the system security.

(2) Libpreopen eradicates tedious application modifications, developers have to make in order to incorporate Capsicum compartmentalization sandbox capabilities in their application.

2.2 Implementation

Libpreopen makes it possible for Capsicum to make system calls and access global namespaces in Capsicum capability mode, by being able to workaround system calls and global namespace access request of applications. Libpreopen opens the directories of the files required applications, store the file descriptors associated with these directories in a storage called po_map. Libpreopen performs these actions and other actions to be discussed in this report in a trusted shell program Capsh

Capsh is a shell program that ensures unreliable applications are sandboxed before execution. More reference on Capsh can be found at `https://github.com/musec/capsh`

When Capsh is given a command to execute an untrusted application, Capsh forks itself into a child process, delegates opening of directories the application requests to make a system call on or access global namespace of their contents to Libpreopen . If Libpreopen successfully opens these directories, Libpreopen stores the file descriptors associated with these directories and the path to these directories in an expandable array as listing 1 illustrates.

po_map has two members dirfd

```
#include <stdio.h>

int main(int argc, char **argv)
{
  printf("Hello world!\n");
  exit(0);
}
```

Figure 2.1: Hello World

2.3 Figures

We can include encapsulated PostScript™ figures (`.eps`) in the document and refer to it using a label. For example, MUN's logo can be seen in Figure 2.2.

Figure 2.2: This is MUN's logo

Figure 2.3 shows a chart of MUN's Fall enrollment from 2005 – 2009.[1] The figure was created using the Calc spreadsheet application of the office suite OpenOffice.org.[2] This figure was reduced by 50%.

[1] From *Memorial University of Newfoundland — Fact Book 2009*.

[2] This office suite can be downloaded at no cost from `http://openoffice.org/`. Unlike other commercial office suites, OpenOffice.org may be legally shared with colleagues and fellow students. There are versions for Linux, Microsoft Windows, Mac OS X and Solaris. Also, unlike commercial offerings, OpenOffice.org does not require activation using registration keys.

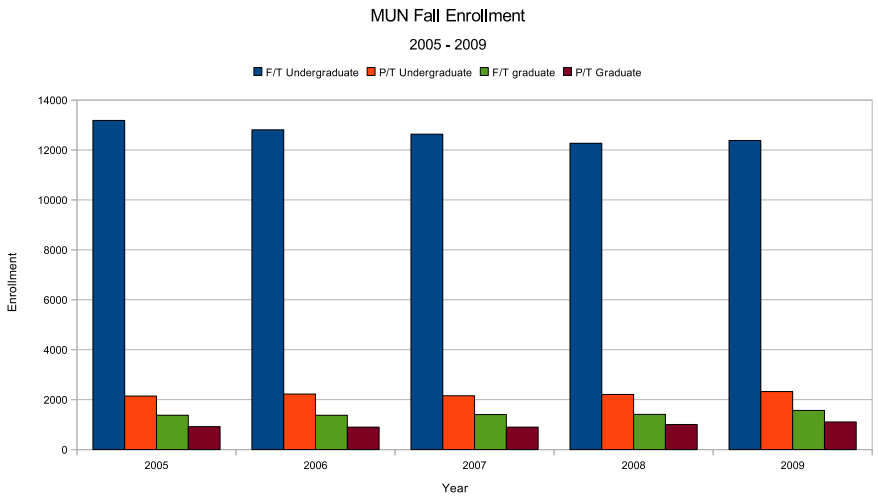

Figure 2.3: MUN Fall Enrollment 2005 – 2009

For larger figures, we can use landscape mode to rotate the page and display the figure using the `\munlepsfig` command, as shown in Figure 2.4. The figure will be the only thing on the page when typeset in landscape mode. (The figure is reduced to 85% of its original size.)

Alternatively, if we just want to rotate the figure, but not the entire page, we can specify an `angle` attribute in the default argument of the `\munepsfig` command. The result is shown in Figure 2.5. If the figure is too large or if there isn't sufficient text, then the figure may appear on its own page.

Note that all three of the enrollment figures are basically the same file, but with different names — on Linux, they are symbolic links to the same file. The filenames have to be different because the reference labels need to be unique.

Figure 2.6 shows a Petri net created using the `xfig` program (`http://www.xfig.org/`) which has very good support for LaTeX. This figure has been reduced to 40% of its original size.

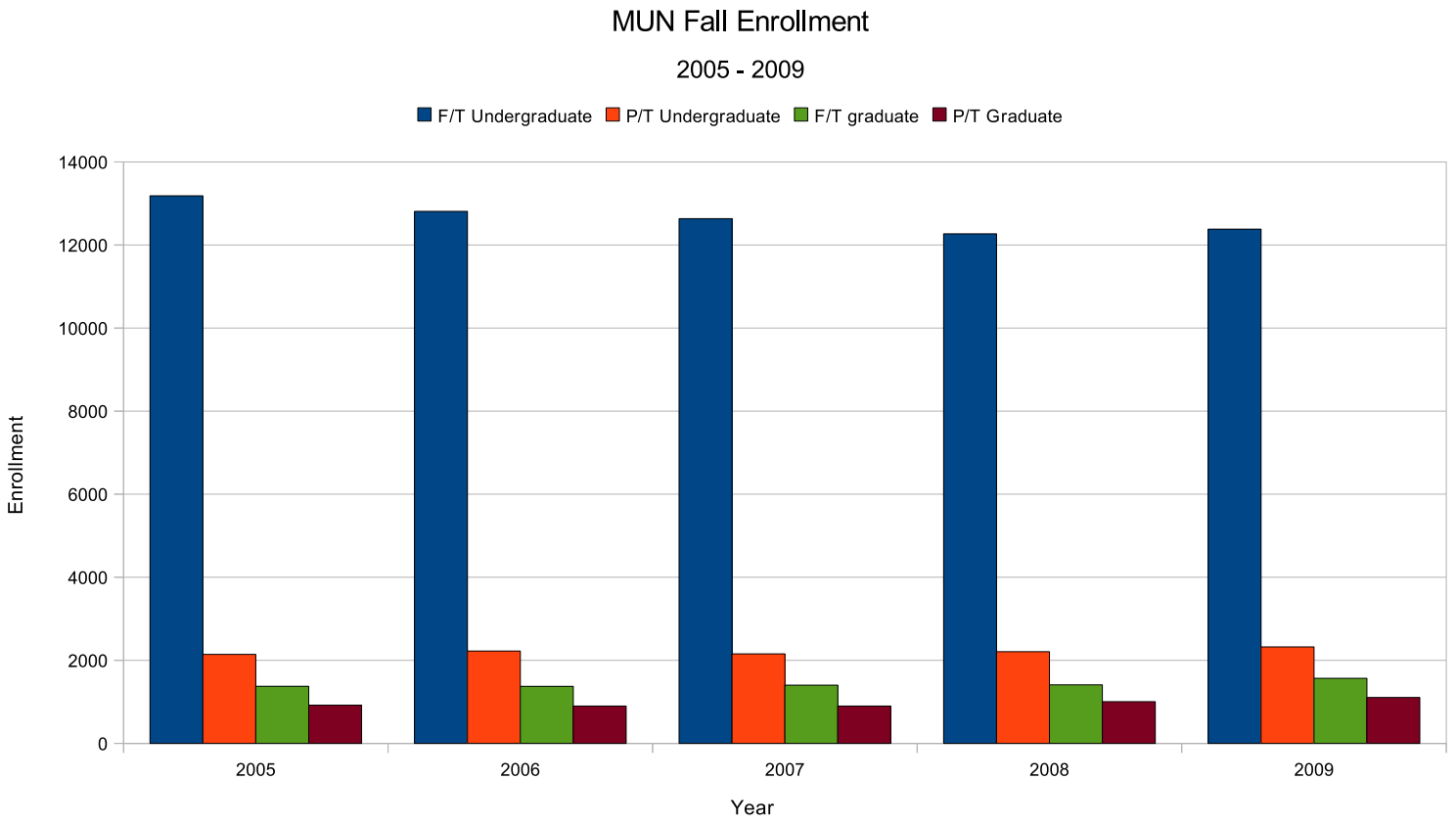

Figure 2.4: MUN Fall Enrollment 2005 – 2009 (landscape)

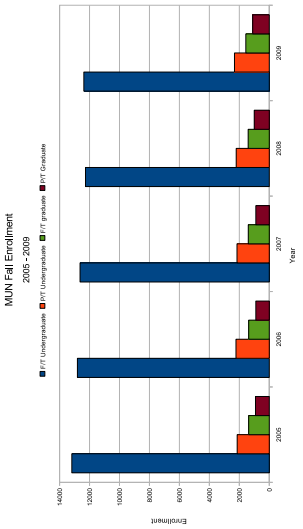

Figure 2.5: MUN Fall Enrollment 2005 – 2009 (rotated)

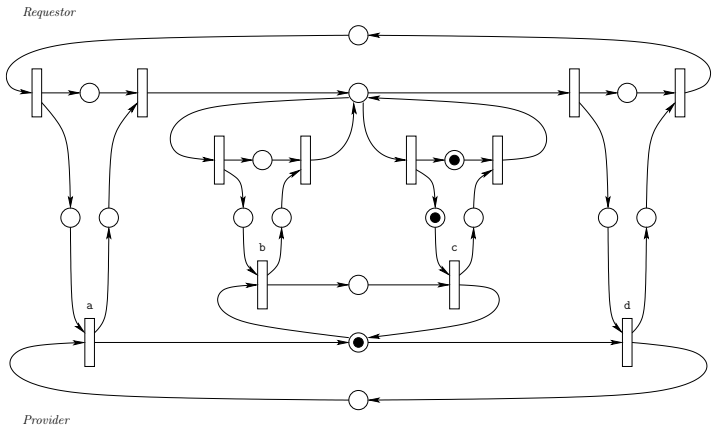

Figure 2.6: A deadlocked Petri net

We can also create figures of text (such as short code snippets) using the `\muntxtfig` command, as show in Figure 2.7.

```
#include <stdio.h>

int main(int argc, char **argv)
{
  printf("Hello world!\n");
  exit(0);
}
```

Figure 2.7: Hello World

2.4 Tables

We can also create tables, as seen by Table 2.1. Note that, as required by SGS guidelines, the caption for a table appears above the table whereas figure captions appear below the figures. Tables and figures can "float" — they may not appear on the page on which they are mentioned. LaTeX tries to handle figure and table placement intelligently, but if if you have a lot of them without a reasonable amount of surrounding textual content, the figures and tables can accumulate towards the end of the chapter. Generally speaking, if there is sufficient text explaining the tables and figures or if the tables/figures are relatively small, this may not be a problem. However, if you have a lot of tables or figures, it may be a good idea to put them in an appendix and refer to them as the need arises.

Table 2.2 shows a different table in landscape mode.[3] This is useful if your table

[3]This data was also taken from the *Memorial University of Newfoundland — Fact Book 2009*.

Table 2.1: Fall Semester Enrollment

	Undergraduate			Graduate		
	F/T	P/T	Total	F/T	P/T	Total
2004	13,191	2,223	15,414	1,308	879	2,187
2005	13,184	2,143	15,327	1,375	920	2,295
2006	12,809	2,224	15,033	1,373	899	2,272
2007	12,634	2,155	14,789	1,403	899	2,302
2008	12,269	2,208	14,477	1,410	1,005	2,415
2009	12,382	2,323	14,705	1,567	1,106	2,673

is too wide for the page. Tables are double-spaced by default. To single-space a table, change the `\baselinestretch` before beginning the table environment. Remember to restore it after the environment has ended.

Table 2.2: Masters Degrees Conferred by Convocation Session — 1950 to 2009

	2009		2008		2007		2006		2006		1950–2004	Total
	May	Oct	May	Oct	May	Oct	May	Oct	May	Oct		
Degrees												
Master of Applied Science	14	2	15	8	28	1	21	3	3	1	98	194
Master of Applied Social Psychology	1	5	2	5	1	4	0	4	0	4	28	54
Master of Applied Statistics	0	0	3	1	0	0	1	0	0	0	19	24
Master of Arts	37	49	26	43	14	42	14	56	13	44	994	1,332
Master of Business Administration	14	16	23	6	33	12	33	11	33	8	818	1,007
Master of Education	107	87	120	55	147	74	108	76	113	75	2,603	3,565
Master of Employment Relations	8	9	5	7	7	14	4	9	3	5	12	83
Master of Engineering	20	19	20	10	16	10	15	13	4	19	440	586
Master of Environmental Science	3	3	3	1	0	1	7	1	3	1	66	89
Master of Marine Studies	2	0	0	1	0	2	2	2	1	2	26	38
Master of Music	4	1	5	0	3	0	3	0	3	0	7	26
Master of Nursing	7	8	10	4	17	4	23	7	6	1	116	203
Master of Oil and Gas Studies	0	0	2	0	0	0	0	2	4	0	0	8
Master of Philosophy	5	4	2	1	5	2	5	3	2	0	112	141
Master of Physical Education	0	2	3	0	5	4	3	0	4	4	84	109
Master of Public Health	0	8	0	0	0	0	0	0	0	0	0	8
Master of Science	40	32	41	19	29	25	35	29	32	23	1,653	1,958
Master of Science (Kinesiology)	1	0	4	2	1	2	2	6	4	3	0	25
Master of Science (Medicine)	18	7	11	8	10	5	9	9	8	4	0	89
Master of Science (Pharmacy)	0	0	1	1	0	0	0	0	1	0	16	19
Master of Social Work	4	11	4	5	4	9	9	5	4	10	257	322
Master of Women's Studies	2	0	2	0	1	1	2	3	2	0	20	33
Total Masters	287	263	302	177	321	212	296	239	243	204	7,369	9,913

Chapter 3

Dealing with Errors

LaTeX can produce cryptic error messages at times. However, with some experience, it is usually not too difficult to determine what the problem is and how to fix it.

As mentioned earlier, appropriate search terms in Google may help you fix these error messages.

Chapter 4

Lorem Ipsum

Now, for your reading pleasure, some *Lorem ipsum*, courtesy of:

<http://www.lipsum.com/>

This gives a good view of the margins — note that the left margin is a bit wider than the right margin to accommodate binding.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam odio elit, viverra eu tempor non, pulvinar ac nisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Sed adipiscing, dui quis viverra facilisis, quam libero adipiscing justo, vitae dictum libero mauris ac magna. Aenean sem ligula, vulputate at vestibulum eu, pellentesque in justo. Sed et eros mauris, sed placerat nulla. Maecenas nulla velit, facilisis et rutrum nec, volutpat id lorem. Duis vestibulum odio velit, id elementum tortor. Sed pellentesque leo ac nibh iaculis at fermentum orci lobortis. Suspendisse arcu magna, porta nec pretium non, feugiat vitae orci. Vivamus at enim arcu, at sagittis nisl. Vestibulum at mi enim, vel malesuada justo. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos.

Nullam sed nunc at enim posuere sagittis. Vivamus augue turpis, mattis a blandit non, sollicitudin non nisl. Integer vestibulum, est vitae cursus adipiscing, elit libero pretium leo, in scelerisque augue felis volutpat nisl. Donec commodo posuere arcu, eget feugiat dui ornare nec. Nullam eros mi, condimentum ac ultricies ac, euismod lobortis nibh. Cras ac ligula pharetra risus elementum pharetra vel in quam. Fusce ac augue vulputate nibh imperdiet convallis sit amet et quam. Integer porttitor dictum fermentum.

Nullam id ante arcu. Nulla facilisi. Vestibulum sodales, mi sodales ultricies pulvinar, orci leo dictum diam, quis imperdiet turpis lacus ut sem. Nulla rutrum odio sit amet elit aliquam blandit gravida nunc placerat. Aenean et neque ut leo condimentum vehicula. Fusce quis orci vitae enim dapibus tincidunt in vel ipsum. Phasellus auctor neque ac eros egestas sit amet ultricies erat vestibulum. Ut erat ligula, pharetra vel hendrerit vitae, mattis ac turpis. Ut malesuada diam vitae lacus vestibulum a tempus nisl posuere. Ut nisi sem, dictum eu laoreet sed, commodo eget enim. Morbi vel lacus neque, tempus fringilla tellus. Nunc id egestas felis. Nullam eu mollis neque. Ut non mauris malesuada eros sagittis congue. Cras vitae felis ut nisl mollis semper ut quis risus. Sed eu arcu urna, et commodo sapien. Donec vestibulum, libero sit amet ultrices blandit, erat lorem volutpat lectus, sed feugiat leo elit in orci. Aliquam vitae leo tellus, placerat pulvinar massa. Nulla at sapien hendrerit diam varius vehicula.

Curabitur et orci nulla. Phasellus euismod, massa non hendrerit dictum, dolor enim imperdiet sapien, vitae commodo lorem tellus eu quam. Duis egestas felis velit. Sed in orci nec nulla rutrum posuere. Suspendisse potenti. Nunc vel quam nisi. In at molestie libero. Aenean hendrerit vestibulum orci, ut hendrerit nulla volutpat lacinia. Vestibulum sit amet sapien vitae lectus gravida vehicula. Suspendisse ac purus sit

amet est congue auctor.

Morbi pellentesque, quam vel mattis molestie, augue purus vestibulum lorem, nec consequat enim eros eu augue. In odio dolor, scelerisque a lobortis porttitor, commodo ut lacus. Maecenas sit amet diam nec tellus accumsan bibendum. Praesent in turpis velit, malesuada commodo sapien. Nunc ornare urna enim. Sed at diam non metus porttitor suscipit. Aliquam erat volutpat. Duis aliquet magna in mauris semper placerat. Ut eget quam orci. Ut egestas, dolor at dapibus accumsan, leo nibh egestas urna, ac consectetur dui odio quis eros. Nam libero dolor, lacinia eget imperdiet non, malesuada vehicula diam. Etiam id ipsum eget turpis consectetur tristique id at ante. Vivamus blandit nunc eu nisl varius sed accumsan odio molestie.

Chapter 5

Handling Citations

BibTEX can be used to handle all your bibliographic needs. Simply add references to the file `ref.bib` and BibTEX will take care of the rest. An example of a BibTEX book, conference paper and journal article are given in the sample `ref.bib` file. Many online journals have links to BibTEX citations that you can download and incorporate into the `ref.bib` file.

The order of the fields is unimportant. BibTEX will display them in the correct order when constructing your bibliography. Also note that you can specify information about a reference that may not even be included in the actual bibliography. For example, the ISBN field is not required by the bibliography, but you can, if you want, put the ISBN to the BibTEX entry.

We can cite a journal article [**?**] and a conference paper [**?**] in the same way as a book citation. More information can be found in [**?**].

Chapter 6

Conclusions

If a vulnerability in an application is exploited and malicious data injected into the system, Capsicum mitigates the spread of the malicious data by con

fining them to the affected process, since an application running in Capsicum sand- boxed mode is compartmentalized into processes and each process sandboxed.

However, an application running in Capsicum sandboxed capability mode is forbidden to access global OS namespaces such as File system, Process IDs, IPC namespaces. The application also has a restricted access to system calls while access to system calls that involves global namespace access is forbidden. In other words, for Capsicum to contain the damage exploit of a vulnerability in an application can cause, the application has to give up its right to perform certain operations.

Applications running in Capsicum capability mode can acquire Capsicum capability rights, and Libpreopen makes it possible for such applications to request system call operations which the applications have the Capsicum capability rights for and have

Libpreopen perform this system call operations with Libpreopen's version of LibC functions.

6.1 Evaluation

Bibliography

[1] A. Ansari. Heap spraying. https://www.exploit-db.com/docs/31019.pdf. Accessed December 16, 2017.

[2] R. N. M. J. A. B. L. K. Kennaway. Capsicum:practical capabilities for unix. *Proceedings of the 19th USENIX Security Symposium*, (3):1–17, 2011.

[3] P. Lonescu. The 10 most common application attacks in action. https://securityintelligence.com/the-10-most-common-application-attacks-in-action/. Accessed December 15, 2017.

[4] Veracode. What is a buffer overflow learn about buffer overrun vulnerabilities exploits and attacks. http://https://www.veracode.com/security/buffer-overflow/. Accessed December 15, 2017.

Appendix A

Appendix title

This is Appendix A.

You can have additional appendices too (*e.g.*, `apdxb.tex`, `apdxc.tex`, *etc.*). If you don't need any appendices, delete the appendix related lines from `thesis.tex` and the file names from `Makefile`.