Welcome & Admin

CYBR473 – Malware and Reverse Engineering (2024/T1)

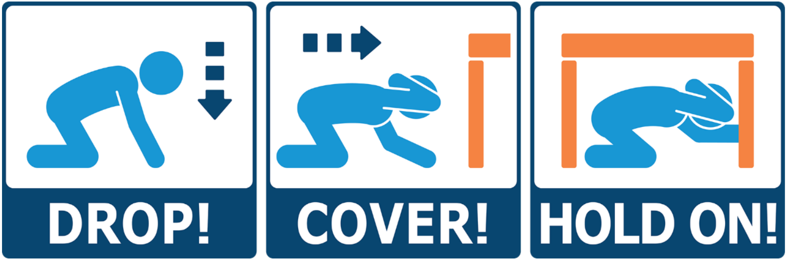
Lecturers: Arman Khouzani (course coordinator), Alvin Valera

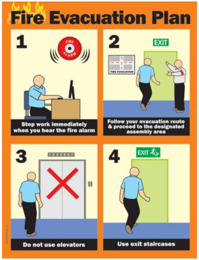
Victoria University of Wellington – School of Engineering and Computer Science

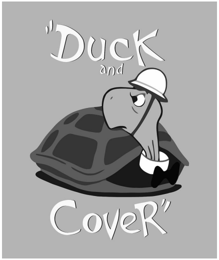
Are you in the right room? This is CYBR473!

Safety Briefing

https://www.youtube.com/watch?v=gUzLLCYeJIM




Fire Evacuation Plan
1
FIRE ALARM
Stop work immediately when you hear the fire alarm
2
EXIT
FIRE EVACUATION
Follow your evacuation route & proceed to the designated assembly area
3
1 2 3 4 5 6 7
Do not use elevators
4
EXIT
Use exit staircases



Teaching Staff

Arman Khouzani, Course Coordinator

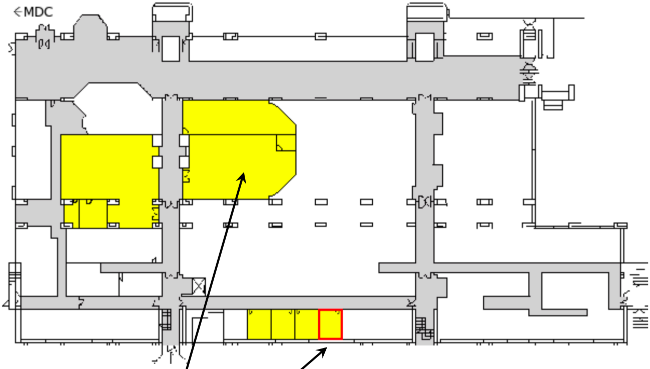
- arman.khouzani@ecs.vuw.ac.nz
- people.wgtn.ac.nz/arman.rezaeikhouzani
- Office: CO129

Alvin Valera

- alvin.valera@vuw.ac.nz
- people.wgtn.ac.nz/alvin.valera
- Office: AM 418

Finding the lab and Arman’s office

Cotton Building (Ground Floor)

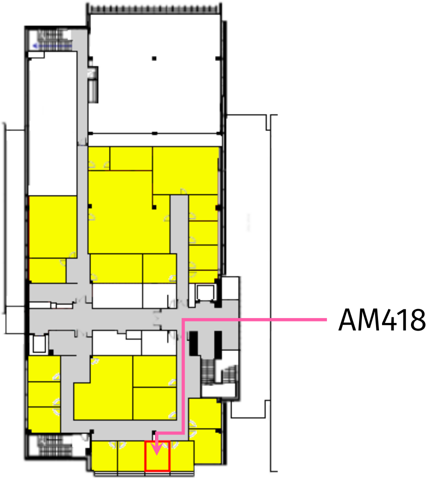


CYBR Lab (CO139)

my office (CO129)

Finding Alvin’s office

Alan MacDiarmid (4th floor)



Course Prescription

This course addresses the problem of identifying and analysing malicious code, using **reverse-engineering** techniques including basic and advance static and dynamic analysis.

Topics will include **methodology** and **techniques** as well as the anatomy, characteristic behaviour of malware.

Practical work will involve malware analysis in a controlled environment as well as the analysis of real-world vulnerabilities and creation of exploits.

Course Learning Objectives

Students who pass this course will be able to:

- Analyse the anatomy, behaviour and propagation methods of malware using **reverse-engineering tools**.
- Detect and bypass attempts by malware to **evade** analysis.
- **Create a proof-of-concept exploit** by applying what you will have learned.

Course Website

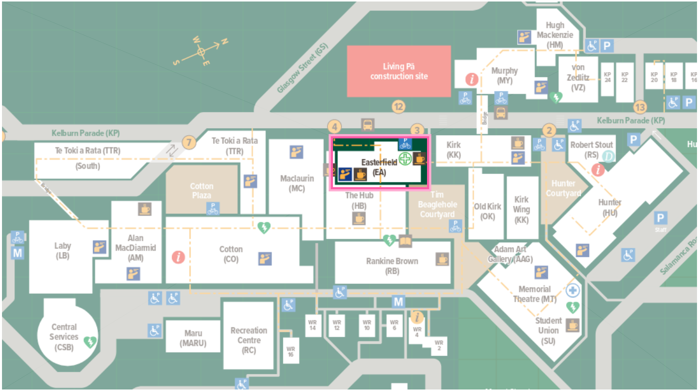
Course Website (ECS wiki):
ecs.wgtn.ac.nz/Courses/CYBR473_2024T1/WebHome

- Course info, slides, reading material.
- Links to lecture recordings (VStream).
- Assignments (times, dates, handouts, files, hints).
- Submission link for assignments.

Announcements via Nuku. **Make sure you check (or forward) your MYVUW email account.**

Course Organisation

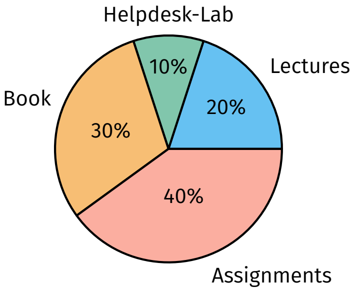
Lectures: **Mon & Wed** @ 11:00-11:50 in Easterfield-101



Helpdesk/Labs: **Fri** 10:00-12:00 in CO139 (CYBR Lab)

Workload (approximate)

- Two lectures per week (2 hours) + Helpdesk (1 hour)
- Reading two book chapters a week = 2∼3 hours
- Working on assignment = 4∼5 hours



15 weeks @ 10 hours per week = 150 hours

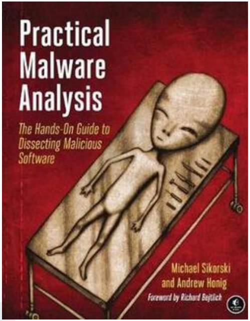
Evaluation Schedule

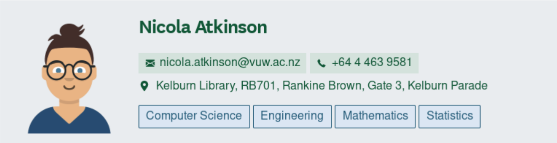
Week	Lecturer	Lab	Assessment
1	Arman	No	
2	Arman	Yes	A1 Released
3	Arman	Yes	
4	Arman	Yes	
5	Arman	Yes	A1 Due (30%)
		No	A2 Released
		No	
6	Arman	Yes	
7	Alvin	Yes	
8	Alvin	Yes	A2 Due (30%)
9	Alvin	Yes	A3 Released
10	Alvin	Yes	
11	Alvin	Yes	
12	Alvin	No	
		No	A3 Due (40%)

Required Textbook

Practical Malware Analysis, by Michael Sikorski and Andrew Honig, 2012, No Starch Press.

- E-book available through library.
- Download PDF of chapters to read offline. **You can also download the entire book as a PDF file.**
- Contact your **Subject Librarian** if you have any technical difficulties:



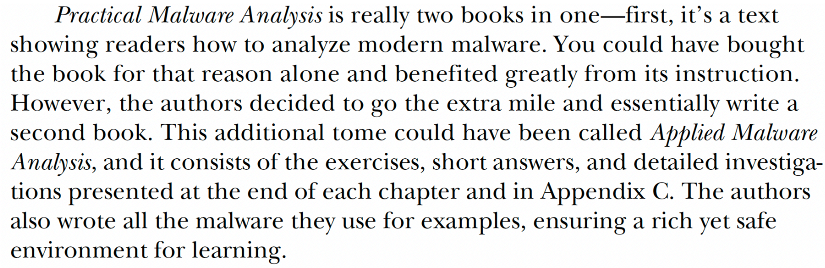


Nicola Atkinson

✉ nicola.atkinson@vuw.ac.nz ☎ +64 4 463 9581

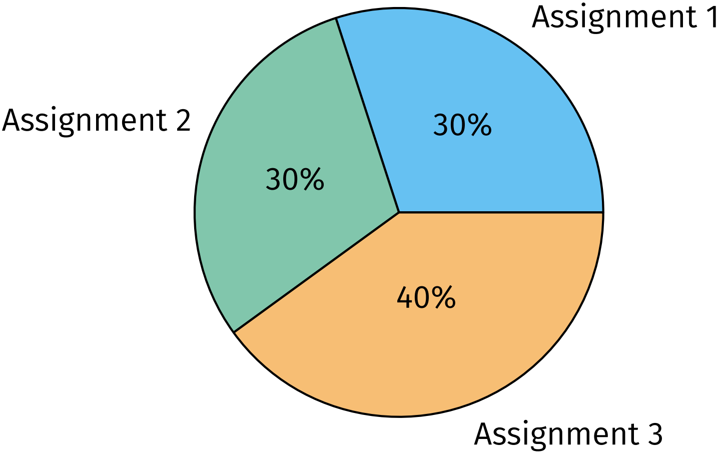
Kelburn Library, RB701, Rankine Brown, Gate 3, Kelburn Parade

Two books in one!



Practical Malware Analysis is really two books in one—first, it's a text showing readers how to analyze modern malware. You could have bought the book for that reason alone and benefited greatly from its instruction. However, the authors decided to go the extra mile and essentially write a second book. This additional tome could have been called *Applied Malware Analysis*, and it consists of the exercises, short answers, and detailed investigations presented at the end of each chapter and in Appendix C. The authors also wrote all the malware they use for examples, ensuring a rich yet safe environment for learning.

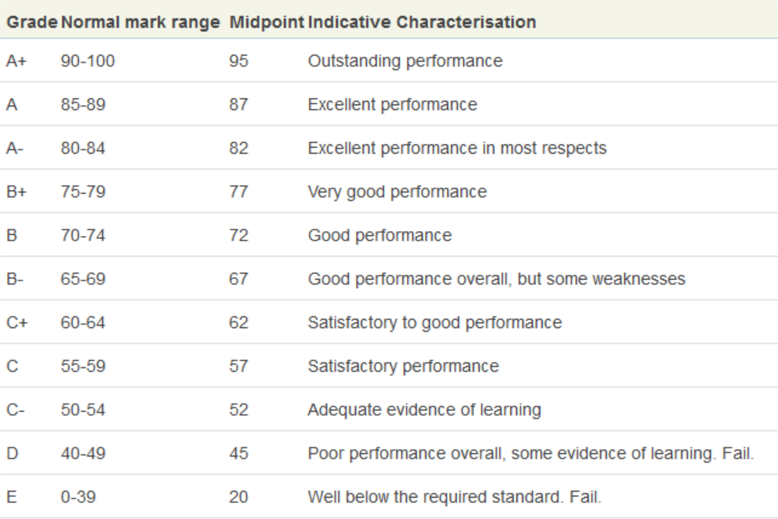
Evaluation Breakdown



Submit through ECS, penalty of **10%** for each late day.

Three 'slip' days available spread over **all** assignments.

Evaluation Grade



Grade	Normal mark range	Midpoint	Indicative Characterisation
A+	90-100	95	Outstanding performance
A	85-89	87	Excellent performance
A-	80-84	82	Excellent performance in most respects
B+	75-79	77	Very good performance
B	70-74	72	Good performance
B-	65-69	67	Good performance overall, but some weaknesses
C+	60-64	62	Satisfactory to good performance
C	55-59	57	Satisfactory performance
C-	50-54	52	Adequate evidence of learning
D	40-49	45	Poor performance overall, some evidence of learning. Fail.
E	0-39	20	Well below the required standard. Fail.

Use of Turnitin

Student work provided for assessment in this course may be checked for academic integrity by the electronic search engine www.turnitin.com.

> *Turnitin is an online plagiarism prevention tool which compares submitted work with a very large database of existing material. Turnitin will retain a copy of submitted material on behalf of the University for detection of future plagiarism, but access to the full text of submissions is not made available to any other party.*

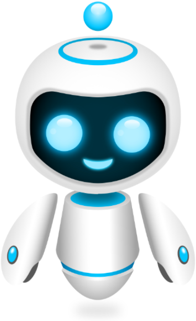
Plagiarism (Cheating): Zero-Tolerance Policy.

You must not present ***anybody else's work*** as your own:

- Basic principle of academic honesty.
- Applies to work by other students, friends, relatives, books, articles, **the web** (blog posts, stack exchange, quora, wikipedia, ...). *Exception:* lecture notes, tutors.
- If you received non-trivial help, then **you must cite it:** *state who helped, and how, and how much.*
- If you **declare** any work of others, then **it isn't plagiarism**, (***but*** they must not have done it for you).

▶ **Zero Tolerance:** Consequences of plagiarism will be severe, include immediate failure of the course.

Plagiarism: AI policy (only for this course!)



AI Orange: You are allowed to use AI tools (**ChatGPT**, **Bing Chat**, **Github Copilot**, **Google Bard**, **Moonbeam**, etc.) to help with coursework **in this course**, ***however***, you must document and cite exactly what you used it for.

Class Representative(s)

> *A class rep is **the bridge** between the lecturer and the students. They are not meant to be a note taker or class life coach, but instead to facilitate feedback by communicating regularly with the class and the course coordinator.*
>
> — VUWSA

Representing your class has benefits: earn points for **Wellington Plus** certificate, professional and personal growth, links to other representation opportunities.

LET'S ELECT NOW!

Big Picture Road Map (Tentative): Part I

Week	Lecture	Topic
1	Mon	Basic Static Techniques
	Wed	VMs, Basic Dynamic Analysis
2	Mon	A Crash Course in x86 Disassembly
	Wed	A Crash Course in x86 Disassembly
3	Mon	IDA Pro
	Wed	Recognising C Code Constructs in Assembly
4	Mon	Analysing Malicious Windows Programs
	Wed	Analysing Malicious Windows Programs
5	Mon	Debugging
	Wed	OllyDbg
6	Mon	OllyDbg
	Wed	Kernel Debugging with WinDbg

Big Picture Road Map (Tentative): Part II

Week	Lecture	Topic
7	Mon	Malware Behaviour (1/2)
	Wed	Malware Behaviour (2/2)
8	Mon	Covert Launching (1/2)
	Wed	Covert Launching (2/2)
9	Mon	Data Encoding in Malware
	Wed	Malware Network Signatures
10	Mon	Anti-disassembly
	Wed	Anti-Debugging
11	Mon	Anti-VM
	Wed	Packers and Unpacking
12	Mon	Shellcode/C++/64-Bit Malware
	Wed	Special Topic: AI for Malware Analysis

Any Questions?