

University of Reading
Department of Computer Science
Daniel Broomhead 27016005
BSC Computer Science
Dr. Mohammed Al-Khafajiy
Logbook

"Investigating the Parallels between
using a RAT-style software for
malicious purposes and virtuous
intent"

Week 1: 21st September 2020

ToDo:

- Set up GitHub Repo
- Start thinking about filling in PID

Week 2: 28th September 2020

Completed Last week: All designated Tasks

ToDo:

- PID

Week 3: 5th October 2020

Completed Last week: All designated Tasks

Due

- PID DUE

ToDo:

- Start Researching Sockets and Networking methods
- Start researching which Languages and platforms to use.

Week 4: 12th October 2020

Completed Last week: started research designated tasks, still in progress,
Continuing further research. → BERKELEY SOCKET API
Compare use cases of implementing on different platforms

Week 5: 19th October 2020

ToDo: Make notes on previous research - in order to aid Report
COVID: Lack of access to resources, limits opportunities and options for
implementation and platforms - due to inability for testing

Week 6: 26th October 2020

DECISION: Programming language → Python
Other languages may be more efficient - but this is effective and useful / Crossplatform
ToDo: Start Researching Socket Implementation and Echo Client
Attempt basic implementation
Simple send / receive, copy, send back using sockets, encoding strings to bytes
Compare methods? UTF8 Looks standard - more research needed

Week 7: 2nd November 2020

ToDo:

Continue previous weeks work,

Investigate connecting multiple clients to a server, or multiple servers to a client,

Work out whether it is necessary for desired implementation.

Would this need listening across multiple ports? Threading?

Week 8: 9th November 2020

Decision:

Multiple connections not needed - not crucial or extremely relevant to intended purpose and design. Possibly last minute extension - but otherwise not.

ToDo:

Continue with previous works, while working on other courseworks

CREATE 3 Environments!: Isolated Testing, Integrated Testing and Live Build

Week 9: 16th November 2020

Other Courseworks

Week 10: 23 November 2020

ToDo: Research Existing solutions and similar properties

Document

Find similarities between existing solutions, what functionalities are key and crucial to an effective implementation. If multiple solutions share functionality or feature -> Then it is either important, useful or necessary -> research which, why and whether mine will need it.

Week 11: 30 November 2020

ToDo:

Plan features and functionality - Doesn't have to be done but make a start and think about what the program could, and should do.

Research Remote Desktops and Virtuous implementations

Week 12: 7th December 2020

ToDo:

Set up git repository properly and learn GIT CMD line

Test importing from github to Gitlab

Research GitIgnore, how to create and what files to add to it.

Keep repository clean

Week 13: 14 December 2020

ToDo:

Investigate reverse shell connections, and shell functionalities,

Reverse shell connections are crucial to a good system as they initiate connection from client side, bypassing most firewall barriers. A good RAT MUST have the ability to use shell commands. Research into what commands can be tested with, and how python handles subprocesses and shell commands.

Week 14: 21 December 2020

Problem occurred: More thorough research into OS.system VS Subprocess required
Os.System() Deprecated? Currently being phased out -> Subprocess much more complicated ->

ToDo: CHRISTMAS

Week 15: 28 December 2020

ToDo: More research in Subprocess: Check output, run, popen, os.system()

When to use each command, when does output need to be returned, when doesn't it?

Think about defining differences between Virtuous / Malicious solutions.

Any famous news stories involving this style of software? Good software gone bad?
Any key

Week 16: 4 December 2020

ToDo: Continue previous Research - unfinished

Research use cases of Virtuous aspects, Malicious is fairly self-explanatory, so why use it for good reasons, and how, limitations, precautions, legal requirements?

Ways of keeping clients safe? -> why trust people with this much power

If you're harmless you're not virtuous, you're just harmless, you're like a rabbit; a rabbit isn't virtuous, it just can't do anything except get eaten! That's not virtuous. If you're a monster, and you don't act monstrously, then you're virtuous.

Jordan Peterson

Quote: Backs up theory, -> Trust is key

How can i promote trust in software

Virtuous Promotes transparency and Trust

Everything is visible and shown to client

Malicious promotes Invisibility,

Client knows nothing, ideally even that the program exists.

Week 17: 11th January 2021

ToDo: Start Implementing Shell functionality,
Isolated testing, integration into abstracted model
Integration into main build

DUE:

FEEDBACK FORM
DEMO TO SUPERVISOR

Feedback:

Very good - seem to be on track, possibly ahead of schedule
Know what i'm doing,
Not worried
Keep working

Week 18: 18th January 2021

ToDo:

Tidy up current workings,
Clean code
Comment code
Continue further research and learning

Week 19: 25th January 2021

ToDo: Focus on other coursework and Deadlines
Keep ticking over

Week 20: 1st February 2021

ToDo:

Start Adding Functionalities
Investigate Shutdown, Restart, Lock, Log off
Different flags

Week 21: 8th February 2021

Problems: More research needed into flags

Possibly to add functions to concatenate and clean all these processes into one?

Benefits of functionalisation?

ToDo: Isolated Testing
Integration into abstracted server / client
More research

Week 22: 15th February 2021

ToDo: Integrate into main program

- Research sending files over socket,
- Start implementing
- Both directions
- Receiving files
- Meeting With SuperVisor: Feedback
 - Looks good
 - Extend features
 - Make sure to describe each in detail in report
 - Start writing report while feature are fresh
 - Session based timer?
 - Acknowledge and disconnect feature?
 - Start thinking about discussion
 - Add an idle timer?
 - Is there a need for a GUI?

Week 23: 22nd February 2021

- Problem: File too large for buffer!! NEED RESEARCH
- ToDo: Respond to feedback
 - Gui -> Not a dominant thought at the moment, can be added last however possibly not even needed.
 - Simple Console UI maybe?
 - More features planned ->
 - Started making notes on feature development to aid report writing
 - Integrate a connect / Acknowledge /Disconnect feature
 - Idle timer ? possible to add but not necessary?
- Continue with file handling

Week 24: 1st March 2021

- ToDo: Research Methods for obtaining System information
 - Work on integrating into isolated model
 - Different methods
- PROBLEMS:
 - Overflowing buffer
 - Printing Dictionaries!! Sending Dict!!
 - Deprecated commands,
 - System Specifics
- More research done into printing dictionaries see DictTest.py

Week 25: 8th March 2021

- Implement solutions found last week, isolated methods, then implement into abstracted model, before integration with main code.
- Tidy code further,
- Heavy Coding week, a lot of design and planning last week to be implemented and tested, integrated and tested, then added to live build.

Week 26: 15th March 2021

ToDo:

- Start researching Keyloggers, Usecases, current solutions, Python libraries, requirements, Clipboard catcher

Make detailed notes to use in the report.

Week 27: 22nd March 2021

ToDo:

- Implement Keylogger, as own class
- Look into threading it onto separate thread
- Research threading python ?
- Start researching how to screenshot in python
- Compare modules for screenshotting

Keyloggers easy to catch with Antivirus due to keyboard hooking, any way to avoid this? -> Do i need to avoid this? → more research needed - clarify reasoning in Report

Week 28: 29th March 2021

Integrate Screenshot -> Isolated, abstract model then live build

Multiple screenshots make a video?

Sleep for framerate? How many seconds, for how long video?

Start researching Telnet

- Telnet Client What can be done with it

- Malicious uses

- Trivial uses, example cases

What is telnet -> What is it used for nowadays ->> Why is telnet client disabled by default -> Alternatives -> reasons to use.

Week 29: 5th April 2021

Turning on telnet client remotely

Integrate all telnet functions from isolated model, to abstracted server/client then into live build

- Capturing keyboard special keys and shortcuts,

- Integrate into keylogger class,

- CTRLC + CTRLV used to combine Keylogger and clipboard grabber

CTRLC / CTRLV are special codes!! How to handle? Not as simple as checking for ctrl and c/v -> must look for escape code /x03

Week 30:12th April 2021

PROBLEM!! : Telnet server that has been running for 15 plus years turned off!!

Contact supervisor

Not much can do -> talk about it, Use screenshots already taken

Mainly proof of concept anyway

ToDo:

Sending files via email

Set up google Account

Set up settings

Email scheduler!!

Get final touches Done !! Integrate everything into live build

Add webcam Functionality and Webcam recording + Playback

Write up more sections

Update Presentation

Week 31:19th April 2021

DUE:

FINAL DEMO

LOGBOOK

ToDo; MAKE SURE EVERYTHING IS DONE

REPORT REPORT REPORT

Feedback for logbook

Finish Logbook

Finish Presentation & Demo

Write up everything.

Supervisor Signature:

Mohammed Al-Khafajiy

Dr. Mohammed Al-Khafajiy

23-04-2021

University of Reading	Department of Computer Science	Daniel Broomhead	27016005 BSC Computer Science		Dr. Mohammed Al-Khafajiy			
Logbook	"Investigating the Parallels between using a RAT-style software for malicious purposes and virtuous intent"							
	Notes	Background Research	Design / Implementation research	Design	isolated Development	Abstracted model integrati	Live deployment	Testing
Folders & Filesending								
Hiding folders		10-Apr	10-Apr	10-Apr	10-Apr	10-Apr	10-Apr	
Buffers		25-Feb	26-Feb	28-Feb	04-Apr	04-Apr	04-Apr	
Sending Files	REDO for Buffer ->	14-Feb	14-Feb	17-Feb	19-Feb	19-Feb	04-Apr	
wireshark								
Receiving files	REDO for Buffer ->	14-Feb	14-Feb	17-Feb	19-Feb	19-Feb	04-Apr	
Video / Webcam / Screenshot								
Video		02-Apr	03-Apr	04-Apr	05-Apr	06-Apr	07-Apr	
Screenshot		27-Mar	29-Mar	30-Mar	31-Mar	01-Apr	02-Apr	
recording playback		16-Apr	18-Apr	18-Apr	19-Apr	19-Apr	19-Apr	
Webcam		15-Apr	15-Apr	16-Apr	17-Apr	17-Apr	18-Apr	
Webcam Recording		16-Apr	18-Apr	18-Apr	18-Apr	18-Apr	19-Apr	
Shutdown / Windows								
Shutdown with message		01-Feb	05-Feb	10-Feb	12-Feb	20-Feb	24-Feb	
Restart		02-Feb	06-Feb	10-Feb	13-Feb	20-Feb	24-Feb	
Log off		03-Feb	07-Feb	10-Feb	14-Feb	20-Feb	24-Feb	
Lock		04-Feb	08-Feb	10-Feb	15-Feb	20-Feb	24-Feb	
Shutdown		01-Feb	09-Feb	10-Feb	16-Feb	20-Feb	24-Feb	
Research								
TCP Vs UDP								
Other Rats								
Feedback								
email to Demo Board	ALL failed to respond :(
email to pat	Useful, templates, on Gdrive							
feedback from academic mentor	in Notes folder:							
feedback from supervisor	in notes folder:							
Halfway Demo	Very good -> go into more depth on TCP							
Demo Feedback	Awaiting							
Meeting with Supervisor								
RESEARCH HEAVY								
Reverse Shell		15-Dec	20-Dec					
MultiConnection		30-Oct	30-Oct	03-Nov	04-Nov			
Os.system vs Subprocess		28-Dec	10-Jan	11-Jan	12-Jan	13-Jan	14-Jan	
Echo Server / Client		29-Oct	30-Oct	31-Oct	01-Nov			
TELNET								
Starwars		01-Jun	01-Apr	05-Apr	05-Apr	05-Apr	11-Apr	
Telnet		01-Apr	03-Apr	04-Apr	05-Apr	05-Apr	11-Apr	
Telnet Client		02-Apr	04-Apr	10-Apr	10-Apr	10-Apr	11-Apr	
Weather		05-Apr	05-Apr	05-Apr	05-Apr	05-Apr	11-Apr	
chess		05-Apr	05-Apr	05-Apr	05-Apr	05-Apr	11-Apr	
clipboard		01-Apr	01-Apr	01-Apr	01-Apr	02-Apr	03-Apr	
Shortcut & special keys		05-Apr	05-Apr	06-Apr	06-Apr	09-Apr	11-Apr	
Keylogger		20-Mar	24-Mar	24-Mar	25-Mar	30-Mar	11-Apr	
dictionaries		02-Mar	03-Mar	06-Mar	10-Mar	10-Mar	12-Mar	
System information		01-Mar	03-Mar	10-Mar	10-Mar	11-Mar	12-Mar	
scheduler		12-Apr	12-Apr	12-Apr	12-Apr	12-Apr	12-Apr	
Emailer		12-Apr	12-Apr	12-Apr	12-Apr	12-Apr	12-Apr	
scripts		16-Apr	16-Apr	16-Apr	16-Apr	16-Apr	16-Apr	
executables		16-Apr	16-Apr	16-Apr	16-Apr	16-Apr	16-Apr	
maze		17-Apr	17-Apr	17-Apr	17-Apr	17-Apr	17-Apr	
pyinstaller		17-Apr	17-Apr	17-Apr	17-Apr	17-Apr	17-Apr	
Deadlines								
PID		09-Oct						
Feedback Form		15-Jan						
Demo to Supervisor		16-Jan						
Poster		26-Feb						
Presentation		20-Apr						
Logbook	*Recursion found*	23-Apr						
Report		29-Apr						
Code		29-Apr						