Chris Turner

s3931227

Github repository: https://github.com/Sevgi-Z/sevgi-z

Github pages site: https://sevgi-z.github.io/sevgi-z/

Chris Turner - My Profile (Intro to IT - Assignment 1)

Contact Details

Name: Chris Turner

Student Number: s3931227

Student Email: s3931227@student.rmit.edu.au

Background Information

As you can obviously tell, my name is Chris, I'm 21 years old and studying a Bachelor of Information Technology.
I was born and raised here in Australia with a dad who came here from the UK, making me half English.

Prior to being at RMIT, I went back to school to do both years of my VCE at TAFE in order to get into university (as I did half of my first semester of high school and dropped out). Although life got in the way near exam time meaning I absolutely flunked them.
This led me to doing a Certificate IV in IT and a Diploma of Software Development at TAFE, and now I'm here. Going on to do computer science would be great.

I was going to put down being half English as my 'interesting fact' but I had to use that for my nationality/culture which makes my interesting fact that I was born, which either says a lot or nothing about me depending on how you look at it.
Although my hobbies count as an interesting fact then my said hobbies would include the standard video games & anime type stuff, and that I enjoy doing 3D modelling.

Interest in IT

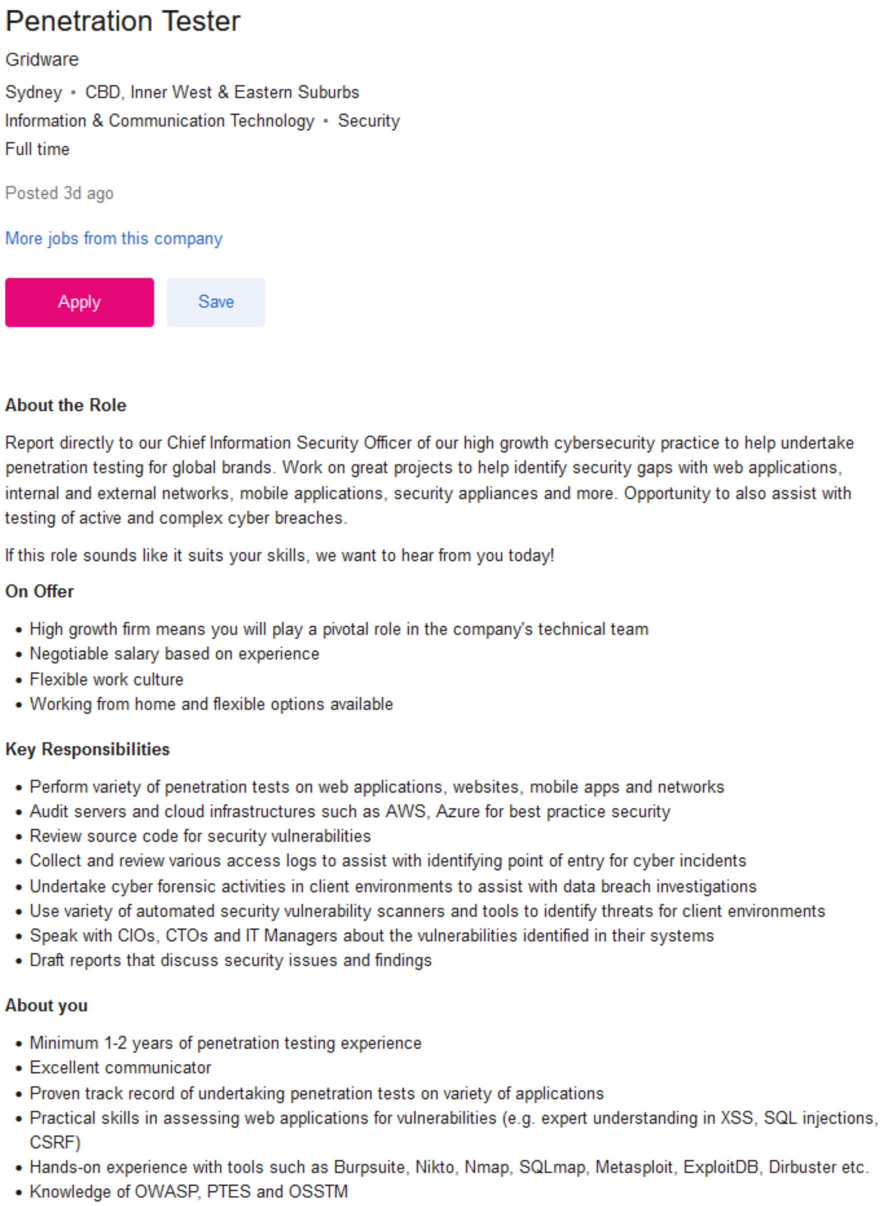
My interest in IT started when I was about 13 or 14 years old. I always enjoyed learning about the insides of computers, what all the parts do, how to take things apart, etc. I was also interested in how software actually works on a technical basis and how it’s created. Despite having an interest so early, my programming skills really aren’t as good as they could be.

There wasn't really something that sparked my interest, but that it’s always been there as I have always been around and used technology. The most IT experience I have, aside from studying at TAFE, is stuff done in my spare time. Fixing and messing around with computers, and learning software development skills and topics in my spare time. I’ve become familiar with C# and Java in desktop and mobile environments.

My interest in IT is pretty broad so It's hard to say if I have a super specific area that I am interested in the most (at the moment), but it would most likely be software development (Windows and Linux environments, game dev or reverse engineering using hex editors and IDA for example), and that I would like learn a lot more about cyber security even if I don’t know much about that area yet (areas of network security and critical infrastructure, as they say)

The main reason for coming to RMIT is pretty much mainly because I couldn't get into Monash. It’s easy to get to I guess? While I'm here I hope to learn not only about software development and cyber security, but many areas that fall under IT as I enjoy learning about as many things as possible.

Ideal Job



Penetration Tester

Gridware

Sydney • CBD, Inner West & Eastern Suburbs

Information & Communication Technology • Security

Full time

Posted 3d ago

More jobs from this company

About the Role

Report directly to our Chief Information Security Officer of our high growth cybersecurity practice to help undertake penetration testing for global brands. Work on great projects to help identify security gaps with web applications, internal and external networks, mobile applications, security appliances and more. Opportunity to also assist with testing of active and complex cyber breaches.

If this role sounds like it suits your skills, we want to hear from you today!

On Offer

- High growth firm means you will play a pivotal role in the company's technical team
- Negotiable salary based on experience
- Flexible work culture
- Working from home and flexible options available

Key Responsibilities

- Perform variety of penetration tests on web applications, websites, mobile apps and networks
- Audit servers and cloud infrastructures such as AWS, Azure for best practice security
- Review source code for security vulnerabilities
- Collect and review various access logs to assist with identifying point of entry for cyber incidents
- Undertake cyber forensic activities in client environments to assist with data breach investigations
- Use variety of automated security vulnerability scanners and tools to identify threats for client environments
- Speak with CIOs, CTOs and IT Managers about the vulnerabilities identified in their systems
- Draft reports that discuss security issues and findings

About you

- Minimum 1-2 years of penetration testing experience
- Excellent communicator
- Proven track record of undertaking penetration tests on variety of applications
- Practical skills in assessing web applications for vulnerabilities (e.g. expert understanding in XSS, SQL injections, CSRF)
- Hands-on experience with tools such as Burpsuite, Nikto, Nmap, SQLmap, Metasploit, ExploitDB, Dirbuster etc.
- Knowledge of OWASP, PTES and OSSTM

Direct Link:
https://www.seek.com.au/job/56124591?type=standout#sol=35f0e7dbbcad6927bb5ef9f5c584805e9e78eec4

The position described is regarded as a "penetration tester", meaning it will be this role's responsibility to discover security flaws. In other words, ethical hacking. With permission, you will "hack" the software or network and report back how you did it so they can be patched. This also includes analysing source code for any security flaws.

Penetration testing/ethical hacking is the main area of cyber security that interests me the most, as this sort of thing can require a low level understanding of the software or hardware being used, which I've always found interesting (the low level aspect). I’ve also always found how security branches occur both through social engineering type tactics, but also on a technical basis (as like in this ideal job), seeing as how cyber crime has become a major concern in the last decade or two, and how it will progressively become even more of a concern.

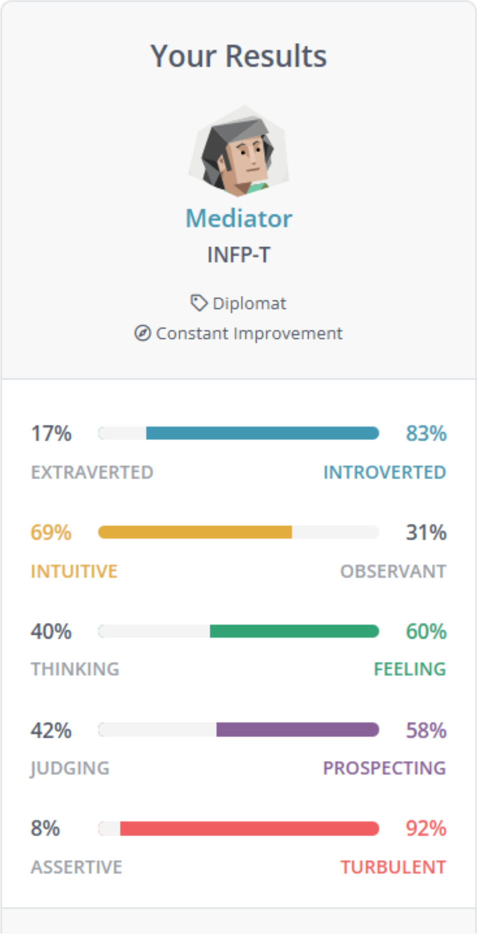
The main skills required for this position and most likely many others like it are:

- Great communication skills, meaning you can work together as a team, clearly discuss plans and relay vital information to your team or the client
- "Practical skills" in regards to penetration testing. This means being familiar with the most common cyber security faults found and having a solid understanding of them (how they're performed and why).
- Knowledge and experience with many pieces of software designed to help people find security faults (SQLmap, Burpsuite, etc)
- Knowledge/understanding of cyber security open standards and methodologies (OWASP, PTES, OSSTM)
- Plenty of real world experience in penetration testing

At the moment, I have practically none of the skills required for this sort of thing, aside from knowing what an SQL injection is, how it's performed and why it happens. Programming skills would most likely be required even if it’s not listed there, which I do have for the most part.
The main ways to actually obtain the skills and knowledge is to: continue my studies in higher education picking relevant courses when possible, gain real world experience most likely through an internship and of course, self study in my spare time.

Personal Profile/Personality Tests

Myers-Briggs/16 Personalities



Your Results

Mediator

INFP-T

Diplomat

Constant Improvement

17% EXTRAVERTED — 83% INTROVERTED

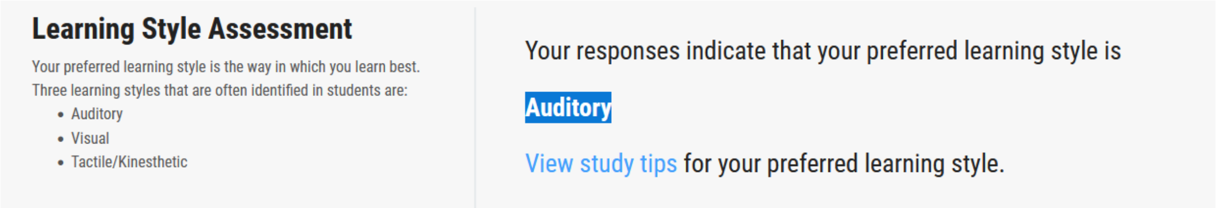
69% INTUITIVE — 31% OBSERVANT

40% THINKING — 60% FEELING

42% JUDGING — 58% PROSPECTING

8% ASSERTIVE — 92% TURBULENT

Learning Style



Learning Style Assessment

Your preferred learning style is the way in which you learn best. Three learning styles that are often identified in students are:

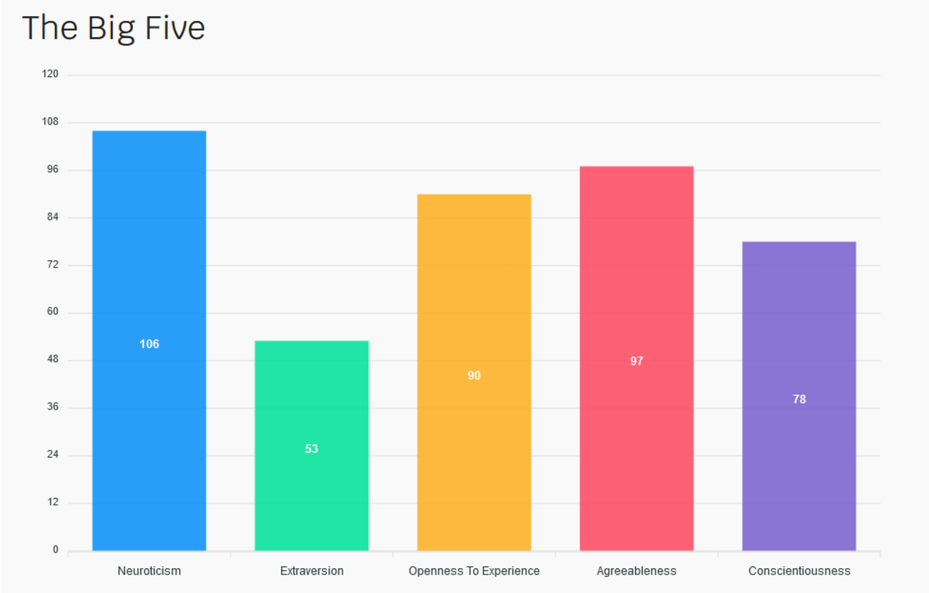
- Auditory
- Visual
- Tactile/Kinesthetic

Your responses indicate that your preferred learning style is

Auditory

View study tips for your preferred learning style.

Big Five



Looking at these results, they are partially accurate and describe some important details/traits to consider when working in a group.

These specific traits are more introverted and perfectionist ones. Working in a team is the opposite of being introverted as it is vital that you clearly and regularly communicate with others about the tasks and problems relevant to the current projectedt (which may not happen if I happen to be introverted and continue to keep to myself).

Being overly critical and having a constant desire to improve means that while I will always strive to achieve excellent results which benefits everyone in the team, it also means that I may spend far too long on small details, leaving more important tasks to suffer and really push that deadline (which would severely disappoint the team, especially if their tasks rely on mine). Having high agreeableness is good as it means the chance of conflict is reduced, but it's important to have critical discussion of your ideas and contributions to get not only consistent effort/quality, but also the best ideas. People should regularly raise questions, criticism and potential problems instead of agreeing with everything.

I neglected to mention the learning style test as it was completely wrong, as I'm more of a mix of all the styles, so I do not have much to say about that part.

IT Project Idea

Video Game via Unity / C#

Project Overview

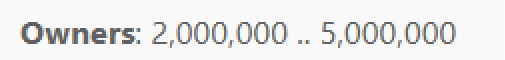
My project idea for the semester is video game built via Unity 5 with C# (3D).

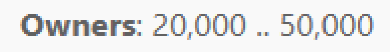
Specifically:

- It will be a more linear survival horror orientated game, with focus on resource management and hints of stealth mechanics
- Able to progress without engaging in combat (allowing for more player choice), with combat focusing more on a ranged perspective.
- The AI of the game will be more attentive to details for the stealth aspects (reacting to player bumping into objects, things falling over and your shadow for example)
- The game will target Windows platforms for now and be runnable on low end machines, or in other words a low fidelity style that of early 6th generation (roughly 10k polys in any given scene)

Motivation

A project like this has the potential to be successful as interest in both big budget, high fidelity and indie low poly survival horror games have seen a sudden rise in interest the last couple of years from a dedicated community. For example, the following are the predicted sales for both a long running big budget franchise and an entirely new IP from a lower budget indie studio.

Resident Evil 8:  **Owners**: 2,000,000 .. 5,000,000 Source: https://steamspy.com/app/1196590

Martha Is Dead:  **Owners**: 20,000 .. 50,000 Source: https://steamspy.com/app/515960

(Numbers provided by the third party service SteamSpy).

The features mentioned also have the potential to attract people as they are fairly original for this style of game (advanced stealth mechanics, no need for combat etc), meaning it may serve as a breath of fresh air. It features features/elements not commonly seen in such a combination. Many games of the genre, while have seen a resurgence, do follow a very similar and same-y formula, which this projects aims to break or innovate upon.

Description

A more complete and detailed list of the defining features are:

- Stealth system - the player will be able to sneak their way past enemies and levels (or engage if they prefer not to). The AI will be able to detect various aspects of the player including: player sight, sounds from the player (footsteps, reloading, knocking world objects over, player shadow, changes in environment).
- Optional combat - to compliment this stealth system, engaging enemy AI will be completely optional. Depending on how often this happens, the AI may alter to adapt (much more sensitive/aware to player aspects)
- Procedural Animation - a more procedural animation system should be implemented for certain animations. Having this sort of system means there can be a more diverse set of animations, helping with visual feedback without the need to manually animate potential actions. This also helps give a more polished feel. A system like this would be done via raycasting to find bone position.
- Scoring system - at the end the game will calculate various factors and statistics that were kept track of throughout the game in order to give you player a score/ranking. This is to encourage replayability meaning players can get more time out of a shorter game.

Tools and Technologies

A few tools will be required for a project like this.

- Obviously Unity will be required. This can be the free version, as even that comes with most of the features you'd need for a small scale project.
- A machine that is capable of running the Unity Development Kit and compiling large amounts of code is also a given. A standard gaming PC will easily be capable of it.
- Visual Studio IDE for writing the C# code. Again, this can be the free or student version. The Unity package/addon would be required.
- A 3DCG program for creating assets like models and animations. Blender will suffice as it is free and close to being on par with industry standard software. A dedicated texturing program would also be helpful (eg: Substance Painter or Quixel).
- An audio editing program to create basic sounds. Audacity will be fine.

Skills Required

Essential skills needed to do a project of this scale and its features would include:

- Knowledge of the Unity development tools. Because Unity is one of, if not the most popular engine out there for independent development, there is an enormous amount of both official and unofficial documentation for everything you can think of. Learning Unity should not be a very significant barrier.
- Intermediate knowledge of C#. Again like with Unity, C# is a popular language that has been around for a while meaning there is plenty of documentation (including from Microsoft themselves). You can learn the basics of a new language in a week or two (up to the basic concepts of object oriented programming). Knowing your way around an average IDE will help (many are similar, so it will extend to Visual Studio).
- Competent at 3DCG tools. Because of the accessibility and popularity of Blender, there are plenty of tutorials for it. Although this may take a little longer to learn compared to other skills mentioned. Because this will be low poly, it will make it much easier to create assets (this includes texturing and animation).
- No "special" hardware will really be required, an average gaming PC will suffice.
- If certain assets can't be created, it is possible to pay for premade ones.

Outcome

If the project is successful, then it may allow for opportunities to further develop the project and form a complete and releasable product. The problem(s) that would have to be solved for it to be a complete product is creating a coherent title that contains a combination of elements not commonly seen and for the formerly mentioned features to be implemented (and for it to be deemed enjoyable by testers). If even the releasable product is successful then it could allow for the resources to create an entirely new game, starting off with a prototype of course (with the assumption that the first completed project would be sold).