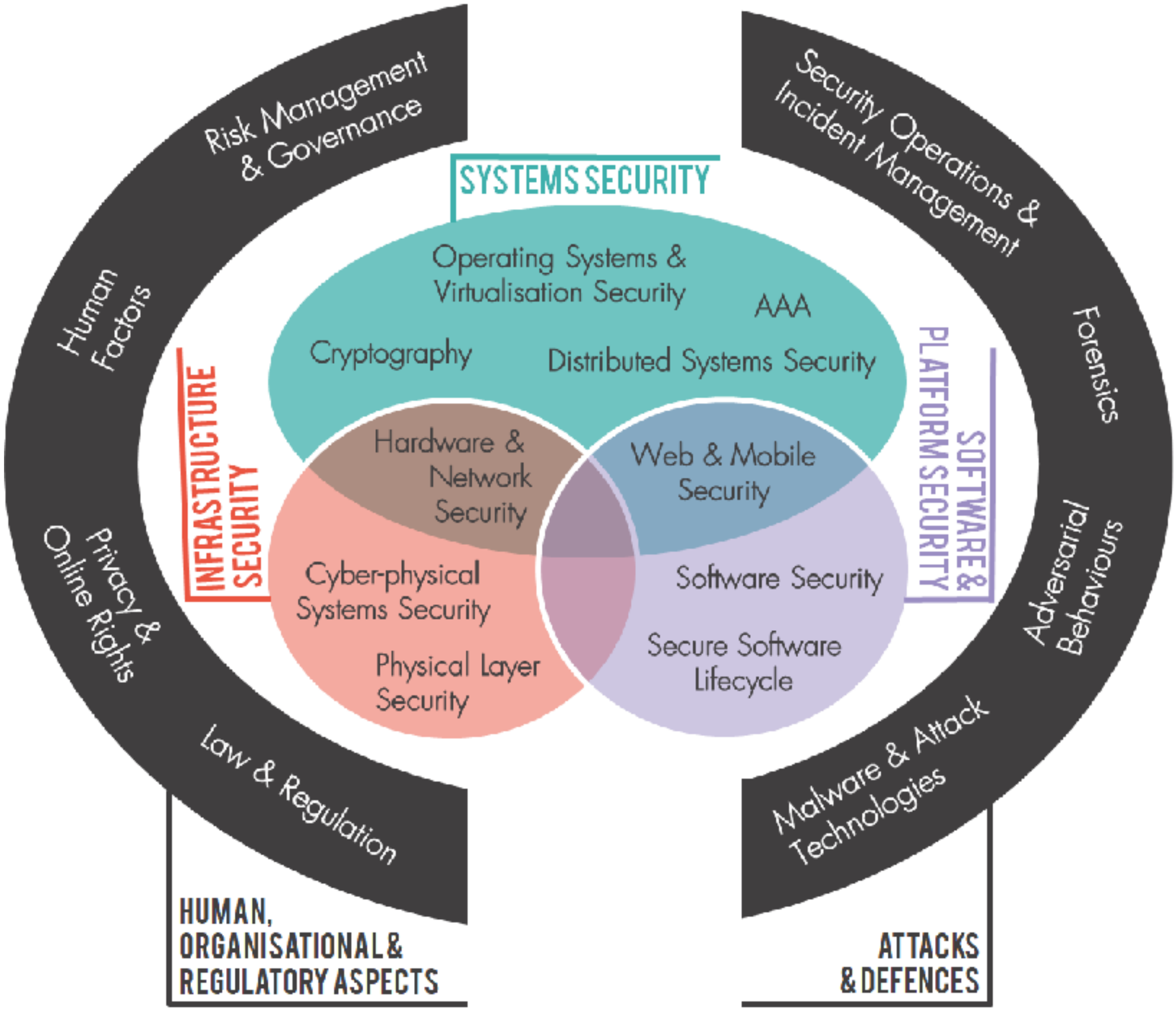
How will you structure research into learning pathways at different education levels?


Risk Management & Governance
Human Factors
Privacy & Online Rights
Law & Regulation
HUMAN, ORGANISATIONAL & REGULATORY ASPECTS
Security Operations & Incident Management
Forensics
Adversarial Behaviours
Malware & Attack Technologies
ATTACKS & DEFENCES
SYSTEMS SECURITY
Operating Systems & Virtualisation Security
AAA
Cryptography
Distributed Systems Security
INFRASTRUCTURE SECURITY
Hardware & Network Security
Cyber-physical Systems Security
Physical Layer Security
SOFTWARE & PLATFORM SECURITY
Web & Mobile Security
Software Security
Secure Software Lifecycle

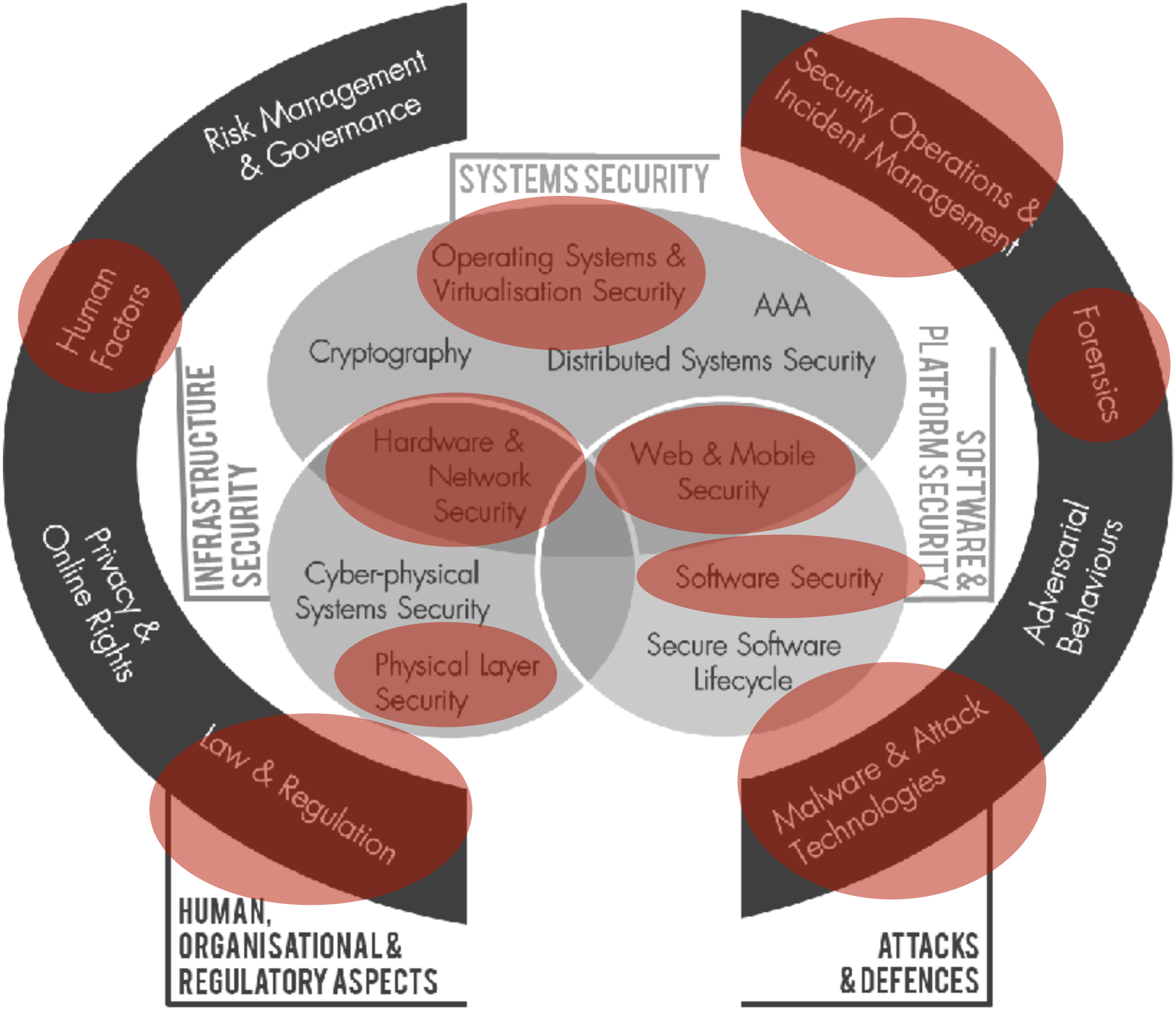
Jobs within cyber-security require knowledge of multiple knowledge areas.

Map the different cyber-security roles to different levels of expertise within each of the knowledge areas.

What knowledge-areas are more general and which are more specialised to specific cyber security jobs?

A survey of different knowledge areas specific to specialist cyber security jobs.

For example a penetration tester needs:



What are the pathways?

Copy the approach of the SWEBOK project.

For each knowledge area produce a matrix linking topics to reference materials.

Pathways are *roughly* the order you go through the different sections of the matrices.

The question becomes what are interesting pathways to map?

	Naik and Tripathy 2008 [1*]	Sommerville 2011 [2*]	Kan 2003 [9*]	Nielsen 1993 [10*]
1. Software Testing Fundamentals				
1.1. Testing-Related Terminology				
1.1.1. Definitions of Testing and Related Terminology	c1,c2	c8		
1.1.2. Faults vs. Failures	c1s5	c11		
1.2. Key Issues				
1.2.1. Test Selection Criteria / Test Adequacy Criteria (Stopping Rules)	c1s14, c6s6, c12s7			
1.2.2. Testing Effectiveness / Objectives for Testing	c13s11, c11 s4			
1.2.3. Testing for Defect Identification	c1s14			
1.2.4. The Oracle Problem	c1s9, c9s7			
1.2.5. Theoretical and Practical Limitations of Testing	c2s7			
1.2.6. The Problem of Infeasible Paths	c4s7			
1.2.7. Testability	c17s2			
1.3. Relationship of Testing to Other Activities				
1.3.1. Testing vs. Static Software Quality Management Techniques	c12			
1.3.2. Testing vs. Correctness Proofs and Formal Verification	c17s2			
1.3.3. Testing vs. Debugging	c3s6			
1.3.4. Testing vs. Programming	c3s2			
2. Test Levels				
2.1. The Target of the Test	c1s13	c8s1		
2.1.1. Unit Testing	c5	c8		
2.1.2. Integration Testing	c7	c8		
2.1.3. System Testing	c8	c8		

Professional Qualifications

You state in the job description that this is an area you're interested in.

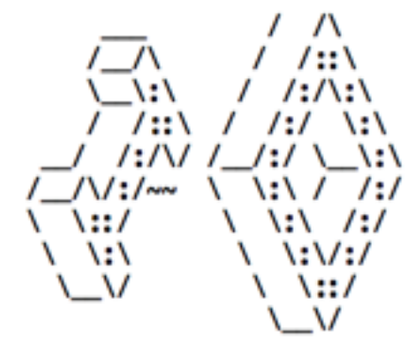
Plenty of qualifications and academic courses available:

- SSCP
- CISSP
- CEH
- Abertay Ethical Hacking course

Less Formal Challenges

What about CTF-style challenge sets?

- IO war game
- Cryptopals challenges
- ROPEmporium



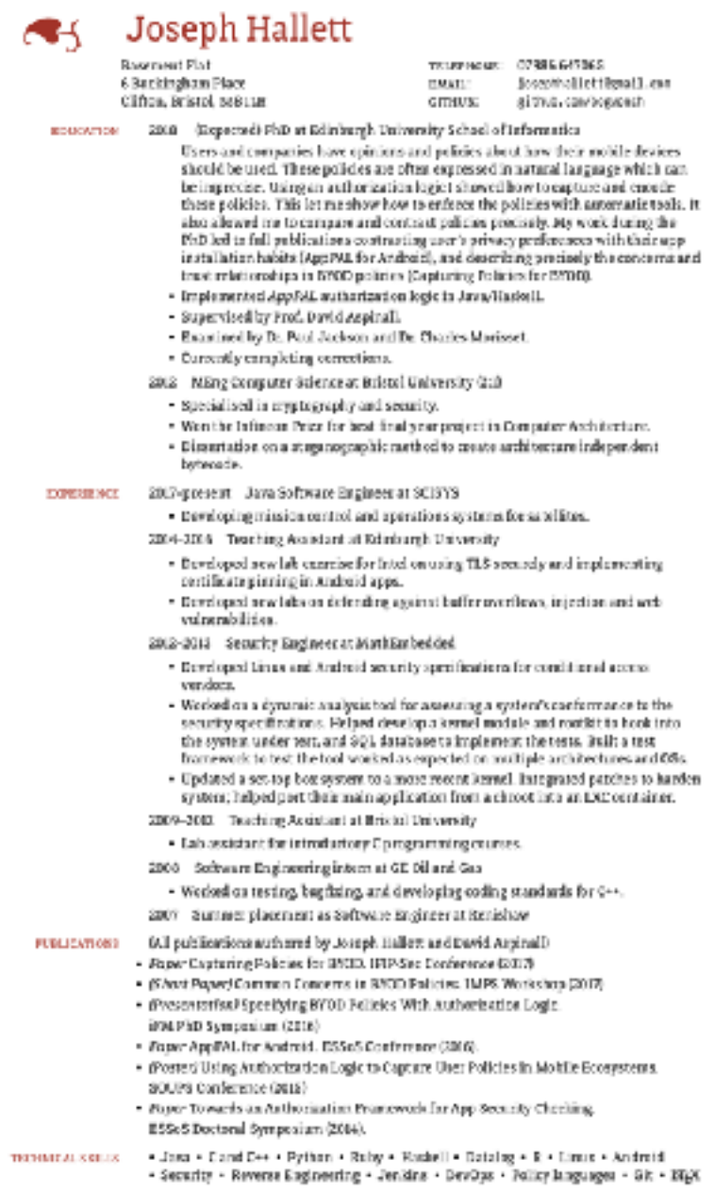
Map solutions to challenges to knowledge areas and show progression through the challenges as pathways

CVs and Jobs

Maps skills and qualifications to developers and companies seeking roles with different levels of experience.

Use required skills to build model for what developers need to know at different stages of their careers.

Estimate of time to progression?



Joseph Hallett

Basement Flat
6 Buckingham Place
Clifton, Bristol, BS8 1LB

TELEPHONE: 07886 647865
EMAIL: joseph.hallett@bristol.ac.uk
GITHUB: github.com/josephhallett

EDUCATION

2010 (Expected) PhD at Edinburgh University School of Informatics

Users and companies have opinions and policies about how their mobile devices should be used. These policies are often expressed in natural language which can be imprecise. Using an authorization logic shows (I) how to capture and encode these policies. This let me show how to enforce the policies with automatic tools. It also allowed me to compare and contrast policies precisely. My work during the PhD led to full publications contrasting user's privacy preferences with their app installation habits (AppPAL for Android), and describing precisely the concerns and trust relationships in BYOD policies (Capturing Policies for BYOD).

- Implemented AppPAL authorization logic in Java/Haskell.
- Supervised by Prof. David Aspinall.
- Reviewed by Dr. Paul Jackson and Dr. Charles Morisset.
- Currently completing corrections.

2008 MEng Computer Science at Bristol University (2:1)

- Specialised in cryptography and security.
- Won the Infineon Prize for best final year project in Computer Architecture.
- Dissertation on a steganographic method to create architecture independent bytecode.

EXPERIENCE

2017-present Java Software Engineer at SCISYS

- Developing mission control and operations systems for satellites.

2014-2016 Teaching Assistant at Edinburgh University

- Developed new lab exercise for Intel on using TLS securely and implementing certificate pinning in Android apps.
- Developed new labs on defending against buffer overflows, injection and web vulnerabilities.

2012-2014 Security Engineer at MathEmbedded

- Developed Linux and Android security specifications for constitutional access vendors.
- Worked on a dynamic analysis tool for assessing a system's conformance to the security specifications. Helped develop a kernel module and rootkit to look into the system under test, and SQL databases to implement the tests. Built a test framework to test the tool worked as expected on multiple architectures and OSs.
- Updated a set-top box system to a more recent kernel. Integrated patches to harden system, helped port their main application from a chroot into an LXC container.

2009-2010 Teaching Assistant at Bristol University

- Lab assistant for introductory C programming courses.

2006 Software Engineering intern at GE Oil and Gas

- Worked on testing, bugfixing, and developing coding standards for C++.

2007 Summer placement as Software Engineer at Renishaw

PUBLICATIONS

(All publications authored by Joseph Hallett and David Aspinall)

- Paper Capturing Policies for BYOD. IFIP Sec Conference (2017)
- (Short Paper) Common Concerns in BYOD Policies. IWPE Workshop (2017)
- (Presentation) Specifying BYOD Policies With Authentication Logic. iFM PhD Symposium (2016)
- Paper AppPAL for Android. ESSoS Conference (2016)
- (Poster) Using Authentication Logic to Capture User Policies in Mobile Ecosystems. SOUPS Conference (2015)
- Paper Towards an Authorisation Framework for App Security Checking. ESSoS Doctoral Symposium (2014).

TECHNICAL SKILLS

- Java • C and C++ • Python • Ruby • Haskell • Datalog • R • Linux • Android
- Security • Reverse Engineering • Jenkins • DevOps • Policy languages • Git • LaTeX

History

New techniques and knowledge hasn’t come from nowhere.

Program stacks lead to *Stack Smashing*, lead to *canaries* and *bounds checking* and *WX* which lead to *ROP* which lead to *RIPROP*.

Map historical progression in techniques as pathways through knowledge areas.

Meta-analysis

If you have the different pathways you can ask questions…

- To what extent are security courses giving developers the skills they need to progress in there careers?
  Compare course pathways to CV pathways. More experienced engineers ought to have completed more of the course pathways.

- How up-to-date is a course?
  Compare course pathways to historical ones. I would expect courses to focus on historical to near recent knowledge areas.

Joseph Hallett

josephhallett@gmail.com