

THE UNIVERSITY OF THE WEST INDIES ST. AUGUSTINE

EXAMINATIONS OF ALTERNATIVE ASSESSMENT 2020

Code and Name of Course: INFO 2604 Information Systems Secu	rity Paper: 1
Date and Time:	Duration: 5 days
INSTRUCTIONS TO CANDIDATES: This paper has 11 pages and 3	questions
WEIGHT 5	50%
Answer ALL Questions	in ALL Sections
Instructions	
 State any assumptions made when answering questions. Answers to questions posted on the forum will NOT students should not ask questions that require the lect questions would be simply to state any assumptions. 	guide students towards the answer and therefore turer to give hints to the answer. The answer to such
PLEASE TURN TO THE N	EXT PAGE
© The University of the West Indies Course	Code INFO 2604 2018/2019
DO NOT WRITE OR TYPE ON THE BACK OF THIS SHE	ET: USE ONE SIDE ONLY
INSTRUCTIONS: Each page must be signed by the Examine the External Examiner. Where the examination does not require a and Second Examiners. Completed forms should be handed to t EXAMINER is requested to sign the question paper and return it w Registrar (Examinations).	University Examiner, the form must be signed by the First he Assistant Registrar (Examinations). The EXTERNAL
- Al oodinge	
First Examiner	University Examiner
Date: 2019 / 5 / 14	Date: 20/

.....

Second Examiner

Date: 2019 / /

External Examiner (where applicable)

Date: 20..../.....



Total Marks = 167

Learning Outcomes being examined

- 1. Apply basic security concepts which are used frequently in the field of information security: confidentiality, integrity, authentication, non-repudiation, authorization and availability
- 2. Investigate mechanisms used for authentication
- 3. Configure a firewall to satisfy security policy goals
- 4. Formulate security policies for a given business scenario
- 5. Demonstrate effective written and oral communication techniques when completing reports and presentations.
- 6. Evaluate cryptographic algorithms used in information security in the context of the overall information technology (IT) industry

© The University of the West Indies	Course Code INFO 2604	2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

- Doodinge	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable)
Date: 2010/ /	Date: 20 / /



Question 1

Part 1

Alice computer (computer A) and Bob computer (computer B) communicates via Protocol 1 which is depicted in Figure 1.

Alice's Computer: Protocol 1

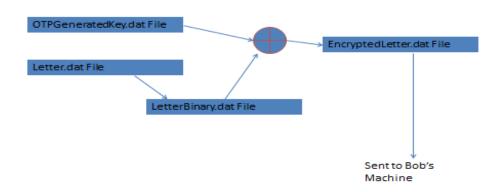


Figure 1
Bob's Computer: Protocol 1

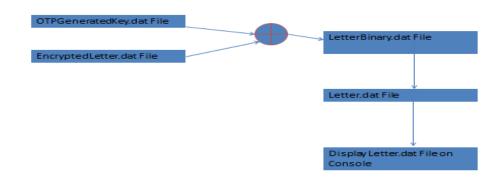


Figure 2

© The University of the West Indies

Course Code INFO 2604

2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

- Al order toge	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable
Data: 2010/ /	Data: 20 / /



Instruction Section

Note 1: Code must be complied with no syntax errors. Also please indicate your Python version and all dependencies used.

Note 2: Proper documentation is required which involves in-code and a *README* file explaining your program code. Full marks would only be awarded if students demonstrate via code comments and short narratives that they understand the code. (*Note 2* is true for ALL code written in this exam)

Note 3: A copy of *Letter.dat* file will be available for download from myElearning.

Requirements

Program A1: Running on Alice Computer (See *Figure 1*)

- 1. Write Python code to generate the key file called *OTPGeneratedKey.dat* which is a file of random 1000 bits (0's and 1's).
- 2. Write Python code to convert the *Letter.dat* file to a binary file containing 0's and 1's and store it to *LetterBinary.dat*.
- 3. Write Python code for the One Time Pad (OTP) encryption function that uses the *OTPGeneratedKey.dat* and *LetterBinary.dat* files and outputs a file called *protocoloneoutput.dat*.
- 4. Write Python code to send the *protocoloneoutput.dat* file to Bob's computer running Program B1 (described below) via socket programming.

Program B1: Running on Bob Computer (See Figure 2)

- 1. Write Python code that listens on a specified port via a socket and accepts the *protocoloneoutput.dat* file.
- 2. Assuming that Programs A1 and B1 run on the same computer from the same directory and that program B1 has access to the *OTPGeneratedKey.dat* file, write a function that uses the *OTPGeneratedKey.dat* and *protocoloneoutput.dat* files and then outputs the result to *LetterBinary.dat* file.
- 3. Convert the *LetterBinary.dat* to an ASCII character file and display the contents of this file to the console.

Note 4: If both Alice and Bob computers share a common drive on the cloud or network makes this scenario possible. But for simplicity assume that Program A1 and Program B1 are running on the same computer.

© The University of the West Indies	Course Code INFO 2604	2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

- Aloodidge	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable)
Date: 2019/ /	Date: 20/



Discussion Question

Discuss two main drawbacks with Protocol 1? Do not include lack of data integrity as a drawback.

Part 2

Using the Diffe Hellman key exchange protocol with p = 7, and g = 5, design and implement a protocol called *ProtocolTransfer* which allows a ProgramA2 running on Alice's computer to send the *OTPGeneratedKey.dat* file to ProgramB2 running on Bob's computer.

Note 5: Both programs A2 and B2 will use different ports from A1 and B1.

Note 6: Use a *DES* or *AES* algorithm (found in a Python Library) to send and receive the *OTPGeneratedKey.dat* file.

Note 7: Using a protocol diagram similar to **Figures** 1 and 2 is helpful along with a description of your protocol.

Part 3

You are required modify part 1 involving Protocol 1 so that an extra piece of information is sent using a Program A3 running on Alice computer to Program B3 running on Bob's computer. This extra piece of information would allow program B3 to ensure that:

- 1. No data alterations were done to the *protocoloneoutput.dat* file.
- 2. That only Alice could have sent the *protocoloneoutput.dat* file.

Note 8: The extra piece of unique data sent to ProgramB3 should be significantly smaller than the *protocoloneoutput.dat* file.

Note 9: Programs A3 and B3 will use different ports from A1/B1 and A2/B2.

Write code for Programs A3 and B3 to check the integrity and authenticity of the *protocoloneoutput.dat* file.

Note 10: Using a protocol diagram similar to Figures 1 and 2 is helpful along with a description of your modified Protocol 1.

© The University of the West Indies

Ast andridge

Course Code INFO 2604

2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

7	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable)
Date: 2019/ /	Date: 20/



Note 11: a README file in your submission is required to show the order to run your programs, and to state whether everything worked, what didn't work, screenshots etc

Rubric

Component				
_	Advanced	Proficient	Approaching	Beginning
			Proficient	
Documentation (10)	Program contains appropriate documentation for all major functions, variables, or nontrivial algorithms. Formatting, indentation, and other white space aids readability. 8 - 10	Program contains some documentation on major functions, variables, or nontrivial algorithms. Indentation and other formatting is appropriate. 5 - 7	Program contains some documentation (at least thestudent's name and program's purpose), but has occasionally misleading indentation. 3 - 4	Program contains no documentation, or grossly misleading indentation. 0 - 2
Modularity Ability to decompose a problem into coherent and reusable functions, files, classes, or objects (as appropriate for the programming language and platform).	Program is decomposed into coherent and reusable units, and unnecessary repetition has been eliminated. 8 - 10	Program is decomposed into coherent units, but may still contain some unnecessary repetition. 5 - 7	Program is decomposed into units of appropriate size, but they lack coherence or reusability. Program contains unnecessary repetition. 3 - 4	Program is one big function or is decomposed in ways that make little sense. 0 - 2
Logic	Program logic is	Program logic is	Program logic is	Program

© The University of the West Indies

Course Code INFO 2604

2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

- Alordin loge	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable)
Date: 2019/ /	Date: 20/



Ability to specify control flow, and data structures that are appropriate for the problem domain. Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results. (20) Syntax Ability to understand and follow the rules of the gold manufall and follow the rules of the gold manufall and glanguage. (10) Ability to specify conditions, who nown boundary error or redundant or contradictory conditions. Brogram produces correct answers or appropriate results for all inputs tested. 10 - 14 Program produces correct answers or appropriate results for most inputs. 10 - 14 Program complies and contain an occasional boundary error or redundant or contradictory conditions. 20 - 29 Program approaches correct answers or appropriate results for most inputs. 10 - 14 Syntax Ability to understand and follow the rules of the gold manufally and the language. (10) Program compiles and contains no evidence of misunderstanding or misunderstanding or superfluous elements. Program compiles and contains an occasional boundary error or redundant or contradictory conditions. 10 - 14 Program compiles and is free from major syntactic misunderstanding or superfluous elements. 10 - 14 Program compiles and is free from major syntactic misunderstanding or superfluous elements. 5 - 9 Program contain an occasional boundary error or redundant or contradictory condition. 20 - 29 0 - 19 Program approaches correct answers or appropriate results for most inputs. 10 - 14 Program contain miscalculations in some cases. 0 - 4 Syntax Ability to understanding or miscalculations or superfluous elements. 5 - 9 Program contain an occasional occasio					
data structures that are appropriate for the problem domain. Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results. (20) Syntax Ability to understand and follow the rules of the programming language. (10) Av - 50 Program produces correct answers or appropriate results for all inputs tested. Program produces correct answers or appropriate results for most inputs. Program produces correct answers or appropriate results for most inputs. Program compiles and contains no evidence of misunderstanding or sunderstand and follow the rules of the language. (10) Program produces correct answers or appropriate results for most inputs. Program compiles and contains no evidence of misunderstanding or superfluous elements. Syntax Ability to undefined non-standard usage or superfluous elements. Syntax Byntax Ability to undefined names Program compiles and contains no evidence of misunderstanding or superfluous elements. Syntax Ability to undefined names	Ability to specify conditions,	,	mostly correct, but may contain an	on the right track	contains some conditions that
that are appropriate for the problem domain. Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results. (20) Syntax Ability to understand and follow the rules of the programing language. (10) Torectness Ability to expect the problem domain. Program produces correct answers or appropriate results for all inputs tested. Program produces correct answers or appropriate results for most inputs. Program produces correct answers or appropriate results for most inputs. Program produces correct answers or appropriate results for most inputs, but can contain miscalculations in some cases. Program correct answers or appropriate results for most inputs. 10 - 14 Program compiles and contains no evidence of misunderstanding or misinterpreting language. (10) Syntax Ability to understand and follow the rules of the language. 10 - 14 Program compiles and contains no evidence of misunderstanding or superfluous elements. Syntax Ability to undefined non-standard usage or superfluous elements. Syntax Ability to understanding or superfluous elements. Syntax Ability to undefined names	control flow, and	errors, and no	occasional		is not correct
that are appropriate for the problem domain. Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for all inputs tested. Corporation and algorithms that reliably produce correct answers or appropriate results. CO) Syntax Ability to understand and follow the rules of the programming language. Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for all inputs tested. Program produces correct answers or appropriate results for most inputs. Program compiles and contain miscalculations in some cases. Program compiles and contains no evidence of misunderstanding or misinterpreting the syntax of the language. Syntax Ability to understand and follow the rules of the language. Syntax Ability to somplies and contains no evidence of misunderstanding or superfluous elements. Syntax Ability to undefined names Program compiles and contain most inputs. Program compiles and contains errors that signal misunderstanding of syntax Syntax Syntax Ability to undefined names Syntax Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for most inputs. Syntax Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for most inputs. 10 - 14 Program compiles and contain miscalculations in some cases. 5 - 9 Program compiles and contain main evidence of misunderstandings, but may contain non-standard usage or superfluous elements. Frogram does correct answers or appropriate results for most inputs. 10 - 14 Syntax Syntax Syntax Program compiles and contain miscalculations in some cases. 5 - 9 Program compiles and contain miscalculations in some cases. 5 - 9 Program compiles or single for most inputs. 10 - 14 Syntax Syntax Program compiles and contain miscalculations in some cases. 5 - 9 Frogram compiles and contain miscalculations in some cases. 5 - 9	data structures	redundant or	boundary error or		
appropriate for the problem domain. (50) Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results. (20) Syntax Ability to understand and follow the rules of the programming language. (10) Program produces correct answers or appropriate results for all inputs tested. Program produces correct answers or appropriate results for most inputs. 10 - 14 Program produces correct answers or appropriate results for most inputs, but can contain miscalculations in some cases. 10 - 14 Program compiles and contains no evidence of misunderstanding or misinterpreting the syntax of the language. 5 - 7	that are	contradictory	J	20 - 29	0 - 19
the problem domain. 40 - 50 Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for all inputs tested. 10 - 14 Program produces correct answers or appropriate results for most inputs. (20) Syntax Ability to understand and follow the rules of the programming language. (10) Program produces correct answers or appropriate results for most inputs. 10 - 14 Program compiles and contains no evidence of misunderstanding or misinterpreting the syntax of the language. 5 - 7 Program produces correct answers or appropriate results for most inputs. or appropriate results for most inputs. or appropriate results for most inputs. Program compiles and contain misunderstandings, but may contain misunderstanding of syntax Program compiles, but contains no evidence of misunderstandings, but may contain mon-standard usage or superfluous elements. 5 - 7	appropriate for	1	contradictory		
domain. 40 - 50 30 - 39					
Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for all inputs tested. 10 - 14 Syntax Ability to understand and follow the rules of the programming language. (10) Program produces correct answers or appropriate results for most inputs. 10 - 14 Program produces correct answers or appropriate results for most inputs, but can contain miscalculations in some cases. 10 - 14 Program compiles and contains no evidence of misunderstanding or superfluous elements. 10 - 14 Program compiles and contains no major syntactic misunderstanding or superfluous elements. 5 - 7	_	40 - 50			
Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for appropriate results. (20) Syntax Ability to understand and follow the rules of the programming language. (10) Program produces correct answers or appropriate results for answers or appropriate results for appropriate results for appropriate results. (20) Program produces correct answers or appropriate results for most inputs. 10 - 14 10 - 14 Program compiles and contains no evidence of misunderstanding programming language. (10) Program produces correct answers or appropriate results for most inputs. 10 - 14 Program compiles and contains no evidence of misunderstandings, but may contain non-standard usage or superfluous elements. Program produces correct answers or appropriate results for most inputs. 10 - 14 Program compiles and contains mo evidence of misunderstandings, but may contain non-standard usage or superfluous elements. Program does not produce correct answers or appropriate results for most inputs. 10 - 14 Program contain most inputs. 10 - 14 Program produces correct answers or appropriate results for most inputs. 10 - 14 Program contain miscalculations in some cases. 5 - 9 Program contain most inputs. 10 - 14 S - 9 Program contain most inputs. 10 - 14 S - 9 Program contain most inputs. 10 - 14 S - 9 Program contain most inputs. 10 - 14 S - 9 Program contain most inputs. 10 - 14 S - 9 Program contain most inputs. 10 - 14 S - 9 Program contain most inputs. 10 - 14 S - 9 Program contain most inputs. 10 - 14 S - 9 Program contain most inputs. 10 - 14 S -	Gorialii.	10 20			
Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for appropriate results. (20) Syntax Ability to understand and follow the rules of the programming language. (10) Ability to code formulae and algorithms that reliably produce correct answers or appropriate results for most inputs. (20) Syntax Ability to understand and follow the rules of the programming language. (10) Ability to compiles and contains no evidence of misunderstanding programming language. (10) Ability to compile and contains no evidence of misunderstanding programming language. (10) Ability to compile sand contains no evidence of misunderstanding or misinterpreting the syntax of the language. (10) Ability to compile sand contains no evidence of misunderstanding or misinterpreting the syntax of the language. (10) Ability to compile sand contains no evidence of misunderstanding or superfluous elements. Ability to compiles and contains no evidence of misunderstanding or superfluous elements. Ability to compiles and contains no evidence of misunderstanding or superfluous elements. Ability to compiles and contains no evidence of misunderstanding or superfluous elements. Ability to compiles and contain major syntactic misunderstanding or superfluous elements. Ability to compiles and contain most inputs. Ability to compiles and contain most inputs. Ability to contain miscalculations in some cases. Ability to compiles and contain major syntactic misunderstanding of syntax Ability to compile or contain most inputs. Ability to compile and contain most inputs. Ability to compile or contain most inputs. Ability to compile and contain most inputs. Ability to compile and contain most inputs. Ability to compile and contain	, , , ,				
formulae and algorithms that reliably produce correct answers or appropriate results for all inputs tested. 10 - 14 Syntax Ability to understand and follow the rules of the programming language. (10) Ability to understand and follow the rules of the language. (10) Ability to understand and follow the rules of the programming language. (10) Ability to understand and follow the rules of the language. (10) Ability to understanding or misinterpreting the syntax of the language. (10) Ability to algorithms that repropriate results for most inputs. 10 - 14 Appropriate results for most inputs, but can contain miscalculations in some cases. 10 - 14 Appropriate results for most inputs, but can contain miscalculations in some cases. 10 - 14 Ability to understanding or misunderstandings, but may contain non-standard usage or superfluous elements. Ability to undefined names			U 1	_	
algorithms that reliably produce correct answers or appropriate results for all inputs tested. 10 - 14 Syntax Ability to understand and follow the rules of the programming language. (10) Ability to understand and follow the rules of the programming language. (10) Ability to understanding or misinterpreting the syntax of the language. (10) Ability to understanding or misinterpreting the syntax of the language. (10) Appropriate results for most inputs. In appropriate results for most inputs, but can contain miscalculations in some cases. 5 - 9 Program compiles and compiles, but compiles, but compiles, but compiles, but compile or contains errors that signal misunderstanding of syntax The appropriate results for most inputs. 10 - 14 Program compiles, but compiles, but compiles, but compiles, but may contain misunderstanding of syntax The appropriate results for most inputs. 10 - 14 Ability to compiles and contains errors in appropriate results for most inputs. 10 - 14 Ability to compiles and contains or major syntactic misunderstandings, but may contain misunderstanding of syntax The appropriate results for most inputs. 10 - 14 Ability to compiles, but compiles, but compiles, but compiles, but may contain misunderstanding of syntax The appropriate results for most inputs. 10 - 14 Ability to compiles, but compiles, but compiles, but may contain misunderstanding of syntax The appropriate results for most inputs. 15 - 9 Ability to compiles and compiles and is free from major syntactic misunderstanding or in a dynamic to understanding of syntax The appropriate results for most inputs. 15 - 9		produces correct	correct answers or		not produce
reliably produce correct answers or appropriate results. (20) Syntax Ability to understand and follow the rules of the programming programming language. (10) Program compiles and contain major syntactic misunderstanding programming language. (10) Results for all inputs, but can contain most inputs. Program compiles and compiles and is free from major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. Program compiles and contains errors that signal misunderstanding of syntax Syntax Ability to undefined names	formulae and	answers or	appropriate results	correct answers	correct
correct answers or appropriate results. (20) Syntax Ability to understand and follow the rules of the programming programming language. (10) Program compiles and contains no evidence of misunderstanding or misinterpreting tlanguage. (10) Syntax Ability to understanding or misinterpreting tlanguage. (10) Inputs, but can contain most inputs. Frogram compiles and compiles and is free from major syntactic misunderstandings, but may contain mon-standard usage or superfluous elements. Frogram compiles program compiles and is free from major syntactic misunderstanding of syntax Frogram contain most inputs. Frogram contain most inputs. Frogram compiles not compiles, but contains errors that signal misunderstanding of syntax Frogram contain most inputs. Frogram compiles not compiles not compile or (in a dynamic language) contains errors that signal misunderstanding of syntax Frogram contain most inputs. Frogram compiles not compiles not compiles not compile or (in a dynamic language) contains typographical errors leading to undefined names	algorithms that	appropriate	for most inputs.	or appropriate	answers or
or appropriate results. (20) Syntax Ability to understand and follow the rules of the programming programming language. (10) Program compiles and contains no evidence of misunderstanding programming language. (10) The program compiles and compiles and is free from major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. The program compiles and compiles, but compiles, but contains errors (in a dynamic misunderstanding of syntax typographical errors leading to undefined names)	reliably produce	results for all		results for most	
results. (20) Syntax Ability to understand and follow the rules of the programming language. (10) Program compiles and contains no evidence of misunderstanding or misinterpreting language. (10) Program compiles and is free from major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. miscalculations in some cases. Program compiles, but contains errors (in a dynamic language) contains typographical errors leading to undefined names	correct answers	inputs tested.		inputs, but can	results for
Syntax Ability to understand and follow the rules of the programming language. (10) In some cases. 0 - 4	or appropriate		10 - 14	contain	most inputs.
Syntax Ability to understand and follow the rules of the programming language. (10) Program compiles and compiles and is free from major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. Program compiles and compiles, but contains errors (in a dynamic that signal misunderstanding of syntax typographical errors leading to undefined names	results.	15 - 20		miscalculations	
Syntax Ability to understand and follow the rules of the programming language. (10) Program compiles and is free from major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. Program compiles and is free from major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. 5 - 7 Program does not compiles, but contains errors (in a dynamic language) misunderstanding of syntax typographical errors leading to undefined names	(20)			in some cases.	0 - 4
Syntax Ability to understand and follow the rules of the programming language. (10) Program compiles and compiles and is free from major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. Program compiles and compiles, but contains errors (in a dynamic that signal misunderstanding of syntax typographical errors leading to undefined names					
Ability to understand and follow the rules of the programming language. (10) Compiles and contains no evidence of misunderstanding or misinterpreting the syntax of the language. Sompiles and compiles and contains free from major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. Compiles, but compile or contains errors that signal misunderstanding of syntax typographical errors leading to undefined names				5 – 9	
understand and follow the rules of the programming language. (10) contains no evidence of misunderstanding or misunderstanding or misinterpreting the syntax of the language. 5 - 7 major syntactic misunderstandings, but may contain non-standard usage or superfluous elements. (in a dynamic language) contains typographical errors leading to undefined names	Syntax	Program	Program compiles	Program	Program does
follow the rules of the programming programming language. (10) evidence of misunderstandings, but may contain non-standard usage or superfluous elements. misunderstandings, but may contain non-standard usage or superfluous elements. follow the rules of misunderstandings, but may contain non-standard usage or superfluous elements. follow the rules or misunderstandings, but may contain misunderstanding of syntax follow the rules or misunderstandings, but may contain misunderstanding of syntax follow the rules or misunderstandings, but may contain misunderstanding of syntax follow the rules or misunderstanding or superfluous elements.	Ability to	compiles and	and is free from	compiles, but	not compile or
of the programming language. (10) misunderstanding or misunderstanding language. (10) misunderstanding or misunderstanding of syntax of the language. 5 - 7 misunderstanding of syntax of syntax of syntax typographical errors leading to undefined names	understand and	contains no	major syntactic	contains errors	(in a dynamic
programming language. (10) or misinterpreting the syntax of the language. 5 - 7 non-standard usage of syntax typographical errors leading to undefined names	follow the rules	evidence of	misunderstandings,	that signal	language)
programming language. (10) or misinterpreting the syntax of the language. 5 - 7 non-standard usage of syntax typographical errors leading to undefined names	of the	misunderstanding	but may contain	misunderstanding	contains
language. (10) misinterpreting the syntax of the language. or superfluous elements. errors leading to undefined names	programming	_	non-standard usage	of syntax	typographical
the syntax of the language. the syntax of the language. to undefined names		misinterpreting		_	
language. names	(10)	the syntax of the	elements.		to undefined
5 - 7		language.			names
8 - 10			5 - 7		
		8 - 10		3 - 4	0 - 2

© The University of the West Indies

Course Code INFO 2604

2017/2018

INSTRUCTIONS: Each page must be signed by the Examiners and where applicable, the University Examiner and/or the External Examiner. Where the examination does not require a University Examiner, the form must be signed by the First and Second Examiners. Completed forms should be handed to the Assistant Registrar (Examinations). The EXTERNAL EXAMINER is requested to sign the question paper and return it with comments, it any, (on a separate sheet), to the Assistant Registrar (Examinations).

- Aloodin lye	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable)
Date: 2019/ /	Date: 20 / /

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY



Parts of Submission	Marks
Programs A1, A2, A3, B1, B2, B3, ReadMe File (Rubric) All program files can be compressed with <i>windows</i> compression into a single file or uploaded separately to myElearning	100
Discussion Question (Word File)	5
Protocol Design for Part 2 (Word File)	10
Protocol Design for Part 3 (Word File)	10
Total Marks	125

© The University of the West Indies	,
-------------------------------------	---

Course Code INFO 2604

2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

- A order lye	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable)
Date: 2010/ /	Date: 20 / /



Question 2

You are given the following "informal firewall policy":

- 1. E-mail may be sent using SMTP in both directions through the firewall, but it must be relayed via the DMZ mail gateway. External e-mail must go through the DMZ mail server.
- 2. Users inside may retrieve their e-mail from the DMZ mail gateway, using either POP3 or POP3S, and authenticate themselves.
- 3. Users outside may retrieve their e-mail from the DMZ mail gateway, but only if they use the secure POP3 protocol, and authenticate themselves.
- 4. Web requests (both unsecured (port 80) and secured (port 443)) are allowed from any internal user out through the firewall but must be relayed via the DMZ Web proxy, which provides authentication and content filtering.
- 5. DNS lookup requests by internal users allowed via the DMZ DNS server, which queries to the Internet.
- 6. External DNS requests are provided by the DMZ DNS server.

Design suitable packet filter rulesets (similar to those done in class) to implement the above policy.

[12 Marks]

© The University of the West Indies

Ast and in dose

Course Code INFO 2604

2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

7	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable)
Date: 2019/ /	Date: 20/

Question 3

The current scheme for doing background checks on school teachers, health service staff and others who have contact with children is too slow. Your job is to design its replacement. The current system has a centralized database and remote sites log into the main server to query the database.

Requirements:

- 1. You are given a database of 20,000 convicted sex offenders stored as (date of birth, name). You may not release any information that might identify an offender. You may only release signed information of date of birth and name.
- 2. Because there are huge peaks in transaction volume at the start of the school year and when National Health Service staff rotates jobs, you want all, or almost all of the digital signatures to be pre-computed for performance reasons.
- 3. The database file is updated daily and provision for constant updates of remote sites is required. Note that database updates only involve appending new sex offenders to the existing database.
- 4. Provision for a poor Internet connection between sites and main server should be considered.
- 5. Each remote site can communicate with the main server via a *network layer security* protocol. Insider attacks are possible and the protocol design has to account for preventing such attacks. This is because packets from the main server traverse a private unsecure network before reaching the router at the main office which in turn sends it on the Internet before reaching the offsite router which in turn sends it through a private unsecure network to the machine with the software that analyses whether or not a person is a sex offender.
- 6. Hackers on the Internet should *NOT* be able to discover any information about the internal structure of private networks.
- 7. There is no possibility of a dictionary attack
- 8. There is no possibility of eavesdropping between server and remote site communication.

Provide an outline *design* for the system and *show* how it meets the above requirements. Note: Designing a system requires outlining the protocols to be used and *how* they are to be used to meet the requirements of a solution. A diagram *may* form *part* of your presentation of the solution. (This diagram will **NOT** be found in the notes – if you decide to use one, it will come from your design) (HINT: Assume that a Certification Centre issued a Certificate for the main server)

[30 Marks]

© The University of the West Indies	Course Code INFO 2604	2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

- Al oodin dye		
First Examiner	University Examiner	
Date: 2019/ 5/14	Date: 20/	
Second Examiner	External Examiner (where applicable	
Date: 2010/ /	Date: 20 / /	



Parts of Submission	Marks
Question 2 (Word File)	12
Question 3 (Word File)	30

END OF EXAMINATION

© The University of the West Indies	Course Code INFO 2604	2017/2018

DO NOT WRITE OR TYPE ON THE BACK OF THIS SHEET: USE ONE SIDE ONLY

- Woodn toge	
First Examiner	University Examiner
Date: 2019/ 5/14	Date: 20/
Second Examiner	External Examiner (where applicable)
Date: 2019/ /	Date: 20 / /