UNIVERSITY OF DUBLIN

TRINITY COLLEGE

FACULTY OF ENGINEERING, MATHEMATICS & SCIENCES

SCHOOL OF COMPUTER SCIENCE & STATISTICS

MSc in Computer Science Hilary Term **2014**

CS7053

Security of Networks and Distributed Systems

CS7053-1

Date	**Location**	**Time**
TBD	**TBD**	**TBD**

Dr. Stephen Farrell

Instructions to Candidates

Please attempt 3 questions.
All questions carry equal marks.

EXAM SOLUTIONS

EXAM SOLUTIONS

Question text is in italics below.
`Answer outlines are like this.`

Question 1. (33 marks)

A general insurance company who market motor insurance and life assurance are putting in place a new sales-support system for their sales staff. The system needs to handle sensitive customer data including financial and healtcare information but also deal with the fact that sales staff are mobile, often visiting customer sites (e.g. large employers whose employees they insure). Avoiding data loss and exposing customer personally identifying information (PII) is a criticial requirement, whilst at the same time ensuring that sales-staff have access to customer records they require to do their jobs. Sales staff will be given a new laptop specially provisioned for running the new system. Internally, the system is a web-based system that is actually operated by a 3rd party cloud provider. At the point where the insurance company have decided to use some 3rd party cloud provider but have not yet identified which one, nor all details of what the 3rd party will be required to provide, you are asked to do an initial risk analysis of their plans.

Answer outlines are like this.

a) Describe the most relevant risks (at least 3) you see, including consideration of their impact and likelihood of occurrence, and outline the countermeasures you would expect that the company and/or 3rd party cloud provider would need to implment to mitigate those risks. (18 marks)

The student should take proper account of the context – insurance, mobile sales staf and a 3rd party cloudy provider. The main point is to describe a reasonable set of threats and mitigations that approximates what might really be used.

The main thing here is to describe risks with their impact and an estimate of probability of occurrence and to concentrate on the more significant of those. The impact and probability can be take any value without losing marks, but for something odd (e.g. if they considered date leakage low impact) they will need more justification for saying that. If the impact/probability are fairly obviously right, less needs to be said. Marking is out of 5 points for each good one and with 3 more for overall goodness. Possible risks would include:

- Leaking of customer personally identifying data and/or financial information
- Data leakage via stolen/lost laptop
- Data leakage via the 3d party
- Denial-of-Service attacks on the 3rd party
- Hackers breaking in to or defacing the 3rd party

- 3rd party staff or other tenants abusing the web infrastructure from within
- Cloud-providers monitoring site traffic and data for their own benefit or for the benefit of competitors or foreign governments
- Staff from other out-sourced providers (e.g. payments processors, customer service) hacking into the system
- Sales or customer service staff working for competitors
- etc. etc.

b) What other questions would you suggest the insurance company ask about the cloud provider's systems and operational processes? (10 marks)

As above basically in terms of marking scheme. This could be more or less focused on security, which is fine, OPS issues are good to ask about too. Possible questions could include:

- Describe your operations security model
- Have you done tiger-team testing
- Describe your risk analysis model
- What SLAs (five-9's etc) do you offer
- What other tennants use this and how to you separate them/us
- How do you report security issues found to your customers (i.e. us)
- etc etc

c) What countermeasures would you put in place to detect or prevent a dishonest member of the sales staff from misbehaving? (5 marks)

As above basically in terms of marking scheme. Dishonest sales folks will either want to sell access to specific records (celebs), to bulk store/sell records to marketers or to bring s/w and records to a competitor. Dectection is going to be far easier, and involve logging/auditing, but logs would need a clever back-end processing setup for stuff to be actually spotted. Prevention mostly based around not letting sales folk have root on the laptop and controlling what the laptop can do, probably via white-lists and not blacklists. Laptops are cheap enough now that there's no reason for sales folk to use the same laptop for other things at all. One tricky on is to try detect if a sales person is using a personal email for customer

interaction, probably too hard but get them to sign that they won't do that and make it a firing offence if they do is a counter.

Question 2. (33 marks)

(a) For any real Internet key exchange protocol (e.g. TLS, S/MIME, IPsec, Kerberos) describe the key management and application data protection aspects of the protocol in detail, including a description of how you would test interoperability between a new and an existing implementaion of your chosen protocol. (15)

Marks-from-10 for a good description of the scheme, with e.g. the TLS handshake and application layer properly described. 5 marks for properly describing a real use case such as IMAP/TLS. Interop testing has to consider corner cases as the most non-obvious important thing, e.g. negotiate use of AES but actually use RC4, and/or trigger recent “goto fail:” cases.

(b) What are the main security failures that are likely to arise in implementations of your chosen protocol? (10)

So much from which to choose! :-(2 marks for each good point made + 2 for overall goodness (expecting 3-4 different points), possibles:

- “goto fail:” cases
- non-constant time crypto
- bad RNGs
- buffer overruns and other standard bad coding
- not checking certs
- not checking revocation
- etc

(c) Desribe and justify how you think your chosen protocol might evolve in the coming deacde? (8)

Most likely here is TLS1.3, where its evolving towards fewer RTTs, getting rid of old/not-so-good crypto, more protection for meta-data where possible etc. For non-TLS cases, evolution will be slower but along similar lines. If they say that a decade is too long to ask about that's a bonus.

Question 3. (33 marks)

You are asked to design a data-retention system for a social networking web site so they can respond to properly validated law enforcement requests for customer information. The web site uses a typical highly scalable web infrastructure with distributed front-end web servers and a distributed database as the backend for the web applications. All customer data that can be accessed is stored in the distributed database. The system needs to provide an interface where specially authorised operators can enter details of the requests received from law enforcement officials including how those requests have been validated. When a validated request is entered into the system a snapshot of all information relating to the customer(s) in questioncan then be copied into a special database. For some requests this snapshot process must be run periodically for some duration defined in the request. For some requests, it is allowed to inform the user of the law enforcement request, but for others the law requires that the request must be kept secret from the user. In all cases, the existence of the request and the associated snapshot data must be handled with extreme care, both from the confidentiality perspective but also in terms of preserving the chain of evidence.

a) Outline the overall design for such a system (include a network diagram) and describe the security requirements the system must meet. (10)

> This should be a fairly straightforward web-like distributed system but with anal-level confidentiality, data-integrity and operator AAA requirements for everything. Interfaces to LE entities may or may not be part of the system. Separating this system from the usual distributed DB (at least as the SQL-level or equivalent) is a near MUST. There could be shared storage beneath though as that might be the best way to get reliability.

b) Describe, in detail, the security solution you would propose to meet those requirements. (15)

> Given a good set of requirements (marking will be lenient in terms of allocating marks to this part for text in the answer to (a)) this is pretty straightforward too, basically use good mechanisms for the confid., integ and AAA requirements. Audit should be part of the solution too though.

c) Having designed the system and being a user of the social network in question, you want to know if you are ever the target for such surveillance. Describe ways in which you might leave in place some form of covert channel so that only you would know if you were being targetted but so nobody else would realise that. (8)

Tricky. But intentionally so:-) Obvious code making a covert channel could be added but should be detected at code review. Some marks for suggesting ideas like that. But the best approach here would probably be to argue that some features (e.g. add then delete a buddy) means that the surveillance subsystem needs to interact with the live system since not all events will always be fully logged otherwise and if you succeed there then make some typical error case (e.g. unfriend a non-existent user) take significantly different durations when done differently (e.g. from different UAs). That'd ideally be done via selecting existing internal APIs that are pretty stable but behave differently so it would not look like a deliberately inserted covert channel even if detected later.

Question 4. (33 marks)

a) Describe the architecture and key management hierarchy of DNSSEC. (15)

5 for arch; 5 for key hierarchy and 5 for overall goodness. This is a straightforward “do they know it” part of the question; they should cover all the basics as taught in class

b) How will DNSSEC deployment impact on registrars, registries and applications? (8)

2 marks for how well they cover apps; 2 each for registrars and registries ops; 2 for overall goodness; points that could be made

- End-hosts will need to have trust points for the DNS root for their resolvers and will need to know DNSSEC
- Some applications will need to include their own resolver and cache in order to be sure that DNSSEC was used (e.g. a TLS implementation in a browser using DANE). Maybe that'll be an interim measure, maybe long term.
- Applications will need to start using some kind of API to tell them if names have been resolved securely.

- Middleboxes (e.g. home gateways) will need to pass DNSSEC results without messing with stuff.
- Registrars will need to include DNSSEC in their “create a domain” Uis and figure out how (or if) to charge for that.
- Registries and authoritative servers will need to do key management including roll-overs and will need to figure out validity periods for signatures (RRsig duration).
- As above for DNS within enterprises.
- ...

c) Describe ways in which the DNS can leak private information and how one might mitigate that with changes to DNS protocols, implementations and deployments. (10)

They should note that DNS data being public is not the same as the act of accessing to that data being public. 3 for covering lack of confid; 3 for covering QNAME minimisation; 4 for overall goodness; -1 if they mix up DNSSEC with this. Bonus is they get the diff between stub and recursive resolver and authoritative server and how that that affects all this and/or things like Google's public DNS servers hoovering up access info.