Question 1 ***Worm Spread*** **(15 min)**

(a) In class we have seen that typical network worms propagate using scanning. Can you think of other ways to spread a worm?

> **Solution:** This question is quite open-ended and has multiple solutions. One solution is to post links to friends via email/Twitter/Facebook. Another is to read logs or other files on the local machine to find other likely victims. A third is to use a search engine ("outsourcing" the scanning). A fourth is to contact a server whose job it is to track other servers (for example, some online games have "meta-servers" that you can contact to find a list of servers available for playing the game).

(b) Bitcoin (and most other cryptocurrencies) use a peer-to-peer gossip network to communicate. In a gossip network, each node has a list of peers. Whenever the node receives a message, it "gossips" the message to all of its peers. This process repeats recursively until the message reaches the entire network–typically within seconds. Why would a memory safety bug in the Bitcoin client's networking code be so deadly?

> **Solution:** An attacker could create a worm, infect everyone's clients and steal everyone's monies.

(c) The typical virus exploits a benign application to execute its own (malicious) code. Exploiting real world applications is getting tougher every year because of the mitigations for buffer overflows that we discussed. Can you think of a way that a virus would not require an exploit to achieve code execution?

> **Solution:** The virus could fool a user to download "critical updates" through social engineering. The *Koobface* worm spread using this technique. Once a user's computer is infected with Koobface, it would post a link on the user's friends' walls. When clicked, the linked page would ask the user to download and install an "update" for their Adobe Flash Player. If the "update" is installed, the computer is now infected with Koobface.

Question 2 ***Abusing Network Monitoring*** **(10 min)**

(a) What does "NOBUS" mean? Give an example from lecture of an instance where the NSA had NOBUS. What are some examples of where the NSA does not have NOBUS?

Solution: NOBUS means "nobody but us": it's the idea that no one else can perform a certain action.

In some cases (like the purported backdoor the NSA allegedly put in Dual-EC) the NSA actually has NOBUS, but more often, the NSA simply uses "off-the-shelf" tools that anyone can use, but use them at a much larger scale than is feasible for the average person (Weaver would say: "More money than God").

(b) Recall that network traffic is often encrypted. Much of the "monitoring" intelligence agencies use is similar to installing NIDS. Given this, how can recording TLS communication still be useful to intelligence agencies?

Solution: Intelligence agencies, like the NSA, often look for metadata.

While the data itself is encrypted, these agencies can still track which websites someone is communicating with, for how long, sending how much data, etc. As we discussed with Tor, often even obfuscation tactics only work if you are the "needle in the haystack": seeing certain types of encryption or security tactics is also important metadata.

(c) What is "cookie linking," and what's its purpose? Be as specific as you can.

Solution: Cookie linking is the process of using the data in cookies to connect traffic (ie, session IDs) to browsers, and ultimately people.

For a specific example: many sites use the same advertising services to host banner ads. Consider site A and site B, which do this. Cookies used for banner ads in both sites would be the same. If site B's traffic leaks any information about the user's identity (suppose this is a social media site), this information can be tracked back to site A's traffic through cookie linking.

Question 3 ***Ransomware*** **(10 min)**

In lecture we discussed *ransomware*, in which the malware holds a computer system or data on it hostage by demanding a ransom for its restoration.

(a) Describe how symmetric encryption could be used for this purpose.

> **Solution:** The virus would have to first encrypt the files, then send the key elsewhere (and delete it locally). The remote party would return the key if payment is made. This works so long as the victim's system doesn't record the key before it is deleted.

(b) Describe how asymmetric encryption could be used for this purpose.

> **Solution:** If asymmetric encryption is used, it would also be possible to generate a key pair ahead of time and embed the public key in the virus while retaining the private key remotely. In this case the user would not be able to decrypt the files even if they record all of the keys present on their system (at any point).

(c) What other types of threats (apart from deleting or encrypting data) can you imagine a virus making to extract payment?

> **Solution:** In 2010, an intriguing virus called Kenzero was out in the wild. Kenzero searches a victim's web browsing history and cache for pornographic websites, takes a screenshot of what it finds, and posts it on a public website with the victim's name. The virus then directs the user to a site where they must make a credit card payment of $16–400 in order to remove the post before Google crawls it.
>
> Perhaps the most interesting possibilities are those that leverage social connections in some way. For example, the virus could find the victim's email contacts or Facebook friends, then threaten to send them something embarrassing. Apart from forwarding existing information, the virus might attempt to make it appear that the victim has taken actions that in fact they haven't (e.g., forge messages or change Facebook relationship statuses). Other possible threats include publicly revealing passwords, secret keys, or credit card information, or consuming resources (e.g., telling other virus instances to constantly send SMS messages to the user's phone).
>
> In order to ensure the user cannot prevent these threats by simply shutting down their machine, the virus could first set up the process so it will be automatically executed by a third-party instance of the virus unless the user makes a payment within some time limit. Once this is in place, the virus would demand payment.

Question 4 ***Trusting Trust*** **(5 min)**

We highly suggest reading article, "Reflections on Trusting Trust."

(a) Think of your daily computer usage, and list down the corporations that you *have* to trust to keep your private data private. Remember that trust is transitive: if you trust EvilCorp, you also have to trust corporations that EvilCorp relies on.

Solution:

This is again an open-ended question. One of the examples we talked about in class was your phone conversation on your iPhone. You have to trust the carriers and the maker of the phone. In addition, you have to trust the corporation that assembled the phone (Apple only designs the iPhone). You have to trust Apple too because it is Apple's software on the phone. In addition, you have to trust the manufacturer of the speaker, the microphone, the camera, the GPS, and so on too.

Interestingly, one of the conclusions that Ken Thompson mentions in his lecture is that you shouldn't trust him or companies that hire him. It's illuminating to go through the list of things you have to abandon if you don't trust Thompson: it gives a clear insight on the massive influence he has had on our daily lives. His Wikipedia page (http://en.wikipedia.org/wiki/Ken_Thompson) is a good place to start.