Question 1:

1. 95% of users are very stupid and don't really care/understand what vendor certified means. If you wanted to trick them you could just add key words like "Official" or "Validated" to the name of your software and the users wouldn't blink an eye.

2. This would certainly help the situation, but from experience, I don't think it would fix the situation. The most effective way would be to enforce this, for example, only allow installations of valid software from certain vendors. Don't leave it up to the users to decide what is safe and what is not. People are going to download software if it looks useful regardless of what the company says. A good example of this is when I downloaded scripts from the internet that told me the weather when I typed "weather". This script could have easily installed malware if it wanted to.

Question 2:

1. A good way to think about DNS is like a phone book, you look up the name of the person and it will give you their phone number. The same thing happens in the DNS "book", you look up the address, like www.google.com and it gives you a IP address. DNS cache poisoning is when you trick the computer into thinking www.google.com goes to a different IP then it really does. So when the user goes to google they might be directed to an attackers site instead of the actual google IP.

2. Predict them poison takes advantage of the fact that only the first port is randomized, after that, they go by increasing ordering until the NAT resets its mapping. For example, if the NAT picks X as its first port, then X+1 will be the next until X+I where I is the number of incoming requests. The idea of predict-then-poison is to first send a few requests to know what ports the NAT is starting at, and then sending spoofed messages with the source IP being the IP of the website that Eve is trying to spoof. Eventually the NAT will start using the malicious packets send by Eve.

Question 3:

1. To stop this, the server and client perform a Diffie-Hellman key exchange to obtain each other's keys. Once this happens only the server can send things that can be authenticated by the client, and vice versa.

2. SSL/TLS will break during the handshake if you have DNS poisoned. The client will ask the server that you have poisoned for a valid certificate, which you can return (this has to be signed by a trusted CA), but then the client will send over the encrypted secret key, and your server will not be able to decrypt it because you don't have the private key. At this point your browser will warn you that something is fishy, and it's up to the user not to be stupid ☺.

3. When first connecting to the server the client will send the version it supports (e.g. SSL 2.0 or SSL 3.0), after this the server responds with its certs and other stuff needed for the handshake. Then the client will send the version number again, “embedded” into the request, and the server will verify that the one sent before matches the one just sent.

Question 4:

Cash: Unlinkable Anonymity

- Cash is completely untraceable. Once you pay for it and leave there is no way for someone to figure out who that cache belongs to.

Gift Cards: Linkable Anonymity

- They will be able to figure out that the same gift card was used to buy things, but assuming you didn't put your name down when buying the gift card, you will still be anonymous. If you are required to give a name, you could give a fake name making it Pseudonymity.

Loyalty card + cash:

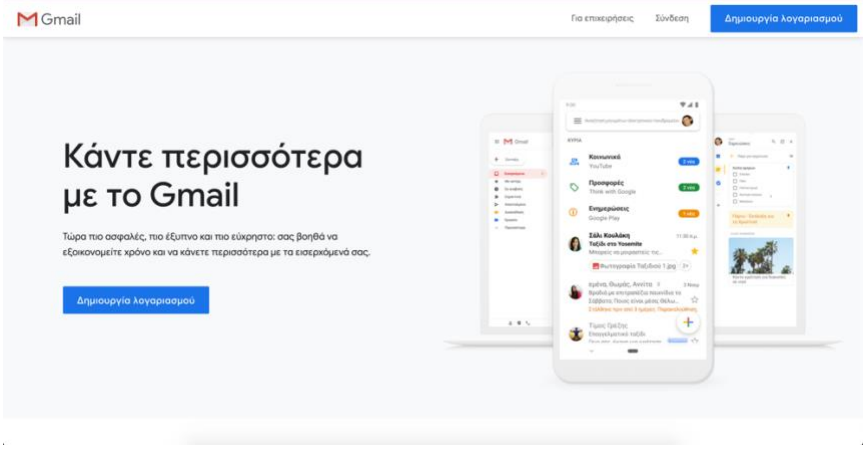
- This one really depends on if they are logging your loyalty card when buying thing. For example, at Walgreens, I put my phone number in so it can find my account and I can get rewards. If this is the case, then this is Verinymity. All purchases can be linked to each other and be traced back to you (assuming you signed up with a real name). This one could also be Pseudonymity if you made an account with a fake name.

Credit Card: Verinymity

- Your credit card is linked to all your purchases and your name is attached to that. Very easy for someone to trace purchases back to you.

Question 5:

1. Three examples:
    a. Searching is around 2-3 times as slow as it is on a regular browser such as Chrome or Safari.
    b. When I went to google and search something I received a warning saying “Our systems have detected unusual traffic from your computer network….”. I have never gotten this error when using normal browsers.
    c. When I went to gmail.com, I was directed to gmail.com in Greek. All the words were in Greek… I attached a picture of the Greek gmail, very fun.



2.

a. The paper talked about how TOR allows for web publishing and instant messaging for services. TOR also allows clients and servers to connect without knowing who each other are. This could be useful for sites like the CIA and new companies.
b. The introduction points are there merely for creating the initial connection for the client and the server. The rendezvous node is there to keep the connection going, although it also does other things. The rendezvous node will be the one that sends the requests back and forth from the server to client and client to server (introduction point does nothing here).
c. Every hidden service should publish their public key so clients can confirm that the public key they were sent is the actual public key of the server. For example, if I was able to DNS poison a site you are going to, I could send you my public key (therefore I also have the private key). If you did not check my public key, you would start encrypting stuff with my public key. At this point I could do anything since I am now trusted by your browser.
d. This allows the client to verify only the server is getting connected to the rendezvous node. If this was not the case, anyone could connect to the node in-between the client sending the encrypted message and the server sending the rendezvous message.

Question 6:

1. XRP
    a. RippleNet is made up of a network of banks and other money lenders/services that use Ripple to provide an easy solution to international lending. XRP is the token-based system that is used on RippleNet. XRP is convertible to other coins, such as bitcoin.
    b. The only major thing that XRP does that bitcoin does not is having a built-in transaction fee, this is not part of the platform you are selling on, this is on the actual coin. Every time you make a transaction .00001 XRP coin is taken away and never replenished into the XRP token pool. They do this to prevent DDOS attacks of transaction, i.e sending billions of free transactions to lag the system.
2. Ethereum
    a. Ethereum is its own platform that allows developers to build software on top of and automatically have safe transactions, no down time, and no third-party company handing transactions. Ethereum is different from most other cryptocurrency's because it is also its own language.
    b. Bitcoin and Ethereum are very different, they serve different purposes. Bitcoin is more for investing and using to buy things, Ethereum is used as a platform for developers to have safe transactions on. People think that it is a lot easy to attack the Ethereum network because the way they do transactions is much simpler, but this also allows for 10-20x speed increase over bitcoin.

Question 7:

1. RSA uses a session key for each session of the RSA servers private key. This allows the server to decrypt incoming requests from clients. If someone got a hold of the session key, they could decrypt all incoming requests. Once the session key changes, they can no longer decrypt messages being sent. Furthermore, if you got a session key, you could not decrypt any messages that were sent in the past, because they are using a different session key.

2. The TLS ECDHE ECDSA with AES_128_GCM_SHA256 mode offers immediate forward secrecy because it creates a new session key with every new connection coming in. Unlike the above question, the same key is not used for every connection. The TLS ECDHE ECDSA with AES_128_GCM_SHA256 mode uses Diffie-Hellman key exchange to negotiate the session key it will use for that connection. This means if an attack got a session key there is very little information that an attack could get.

Question 8:

1. Domain Validation
    a. First, the client must ask Let's Encrypt what it does to have to show that it owns the domain it is querying about.
    b. The client will complete one of the tasks Let's Encrypt sends back to the client to prove they own it
    c. If the CA sees that the task has been completed with expected results, it will allow the client to do certification management.

2. Certificate Revocation
    a. The agent will sign the Revocation message with its key-pair, then Let's Encrypt will verify the request was authorized.
    b. Let's Encrypt will then post the Revocation will publish the message onto revocation channels.
    c. Now when browsers check the certification they will know it has been revoked.

Steven Dellamore

Question 9:

1. Salting passwords adds another layer of protection from the attacker. Salting a password is when you pick a random string, concatenate it with the original password, use the hash function on it and then store it in your database. Another advantage of salting a password is it protects the original password from attackers.

2. Password Managers are made up of three different parts
    a. How Password Managers work
        i. Core service
            1. The part that executes each request in the active directory request by the web service (client). Communications via encrypted channels.
        ii. Web service
            1. The website that the client can log onto and change settings if they would like. Also contains things like a FAQ and a help center if the client gets stuck.
        iii. Local password client
            1. Allows users to manage passwords locally. This allows them to not have to login to the password manager service to access their passwords.
    b. LastPast
        i. Advantages
            1. Supports (almost) every browser
            2. Keeps all your passwords in one place, this makes it so you don't have to go around looking for the eight different passwords you use for different sites.
            3. All passwords are encrypted, hashed, and salted when being stored. Not even LastPast knows the password you used once it has been encrypted.
        ii. Disadvantages
            1. Has a higher risk of being attacked since lots of people use it, and it is supported on every browser.
            2. If someone gets into your passwords, they have everything. Instead of just getting one password, they will have bank, facebook, email, ect ect.

3. Once example of two-factor authentication that I just ran into was when I was trying to reset my password for my bank account. This was more like 10-step authentication but follows the same principles. I had to provide my SSN, birthday, credit card number, and was sent a text with a pin number to verify it was my phone.