# Transcript: IT Podcast - Ep 105 - C836 Lesson 2 - with Arthur Moore and Jessica Galterio

*The following transcript is a verbatim account of the video or audio file accompanying this transcript.*

Speaker # 1: Arthur Moore

WGU's IT Audio Series, flexible, portable, profound.

Hey, this is Arthur, one of the CIs for C836, Fundamentals of Information Security, and this is Lesson 2. Let's get to it. Just as a reminder, this is not meant to replace the learning materials, but only meant to supplement. Even though you can't, it is highly recommended that you do not schedule your first attempt until all the material is covered. Take the pre-assessment, review lessons 1 through 12, pay close attention to the Chapters 1 through 6, complete lessons 1 through 12 flashcards to 100 percent. Again, playing close attention to Chapters 1 through 6, complete the lessons 1 through 12, and test and learn mode, scoring 100 percent, again, running theme for that 52 percent of the test, pay close attention to Chapters 1 through 6.

Identify and authenticate. To identify is to claim who or what we are. Authentication is the act of proving who or what we claim to be. For example, a username is an identity claim and the password authenticates that user and allows access. Identity verification is the half-step between identity and authentication.

Think of it as when you're asked to show a credit card, debit card, and then a government issued ID. That's not really authenticating you. That's just identity verification, which is an additional step of asking for another form of ID. There are five different types of authentication. There's something that you know, username, password, PIN. Something that you have, ID badge, swipe card, one-time password. Something that you are, fingerprint, iris scan, retina scan. Somewhere that you are, geolocation, IP address location, coordinates. Something that you do, that's handwriting, typing, walking because we all do these things, all three of these examples in a slightly different way. It could be the way that you do your O.

It could be the amount of time that you use between each key when you're typing or when walking, how long your strides are.

To go along with authentication, there are different levels of authentication that we can add onto our access controls. We have single, dual, and multifactor. Single is just using one of the factors from the previous slide by itself. Dual factor is using two different factors. If you have a username, password, and a PIN, that does not count as dual factor, that is single factor used two-time.

But if you have a username, password, and then a smart card, that's two different factors. That's dual factor authentication. Multifactor is when we have three or more factors. If we have a username, PIN, and then another PIN that's still just single factor, that two PINs, username, password, that's still single factor just being used three times.

Or let's say we have a username, password, PIN, and then a one-time password, that's dual factor authentication, again, just being used three times. A true multi-factor situation is when we have a username, password, something we know, a smartcard, something that we have, and then a fingerprint scan, something that we are. That is a true multifactor situation, where we have all three factors that are using three different forms of authentication. Now, mutual authentication is not the same as multifactor authentication. Multifactor is stacking those layers on top of each other to make sure that the axis is as strong as we need it to be. Mutual authentication is the process where both ends authenticate one another instead of, for example, just the endpoint or the user authenticating to the server that they're logging in to. For example, both the PC and the server authenticate to each other before data is sent in any direction. The PC authenticates server with a username and password, and in-kind in the opposite direction, the server authenticates to the PC, sending that. Mutual authentication can also be mixed with multifactor authentication because they are two different items.

Remember, multiple factors to access one resource. Be it's something you have, something you are, something you know. Mutual authentication is when both endpoints authenticate each other. We have mutual authentication to prevent man in the middle attacks where as a bad actor or attacker can insert themselves into the traffic flow and compromise confidentiality and possibly integrity of the event because they're in the middle and they can actually see the full communication flow. Passwords, something that you know. Passwords are the most common form of authentication. Yes, we all have about 1,800,000 of them. Or at least you should have more than just one password that you use on multiple systems. I'll hit that in a second here, that's called manual synchronization.

Passwords are very vulnerable if created without complexity requirements. Complexity requirements include numbers, letters, uppercase, lowercase special characters. It's also referred to as camel casing. But you want make sure that you have complex passwords because complex passwords are the countermeasure against brute force attacks when you have a process that's just trying to guess all possible combinations.

If you're using dictionary words, it makes it really easy for brute force attackers to just guess which passwords you have. Password managers are programs that store all the users passwords in one case with master password. Think of it as a skeleton key that unlocks the key for all the other keys, or a master key that unlocks the password for all the other passwords.

Just to touch on what I said earlier, manual password synchronization is when a user synchronizes password from multiple systems without a software application. This should not be done. Yes, it's convenient. Yes, it makes it easy, but a breach for one could end up being a breach for all based off of one bad password.

Biometrics. Biometrics are authentication factors that use physical features. Biometrics are defined by seven different features. Universality, uniqueness, permanence, collectability, performance, acceptability, and circumvention. Universality, it stipulates that we should be able to find our chosen biometric characteristics and the majority of the people that we expect to enroll into the system. Uniqueness is how unique the particular characteristic is among the individuals enrolled. Permanence is how well the characteristic resists change over time. Collectability, how easy it is to accurately collect the characteristic. Performance is a set of metrics on how well to judge the system focus.

Acceptability is the measure of how a particular characteristic is acceptable to the users for the system, and circumvention is how easy it is to circumvent or falsify the biometric information. I want to give an example of these seven. For example, let's talk about a fingerprint reader. Fingerprint readers are easy to install. They're unique. They're universally accepted.

They do persist over time with little change. They're easy to collect. Performance wise, there is going to be a storage issue with having all those there. Acceptability is also easy for the users to access the system with that. Circumvention is, it could be easier to circumvent a fingerprint reader compared to that of a, I'm going to go on the opposite end of the spectrum and go really extreme, let's say, a blood sample. Now it's user universally acceptable because everybody has blood, it is unique. Your blood is unique to you because it has your DNA. It might be very hard to give a blood sample every time you want to access the system. Blood doesn't usually change over the course of a long period of time. As far as performance goes, nobody really wants to give a blood sample every time that we want to access the system.

This is where this particular example breaks down. You can see when you're dealing with biometrics, it has to be universally acceptable and not that easy to circumvent. But more importantly, it has to be acceptable for your users. Because, quite frankly, nobody wants to give a blood sample just to check their ATM machine.

Hardware tokens. Something that you have. Hardware tokens are physical devices that generate a one-time password, and when I mean a physical devices, the device that you actually hold, you press the button, and it gives you your OTP, one-time password. Software tokens are applications that generate one-time password, and that can be on your phone, computer, or any other device where the software is actually doing it.

Just as I said, one-time passwords are passwords that expire after a set timeframe or one-time usage. I am going to bring this session to a close. Thank you so much for coming to this. My name is Arthur Moore. I look forward to serving all my students. Thank you.

Schedule time with your course instructor to explore more deeply. WGU, a new kind of you.